Chapter 12

CNC Program Codes: G-codes

G-code		Group	Description
G00	*	A	Rapid Positioning
G01		A	Linear Interpolation
G02		A	Circular or Helical Interpolation CW
G03		A	Circular or Helical Interpolation CCW
G04		B	Dwell
G09		B	Decelerate and Stop *(formerly known as Exact Stop)*
G10		B	Parameter Setting
G17	*	C	Circular Interpolation Plane Selection XY
G18		C	Circular Interpolation Plane Selection ZX
G19		C	Circular Interpolation Plane Selection YZ
G20	*	K	Select Inch Units
G21		K	Select Metric Units
G22		O	Work envelope on
G23	*	O	Work envelope off
G28		B	Return to Reference Point
G29		B	Return from Reference Point
G30		B	Return to Secondary Reference Point
G40	*	D	Cutter Compensation Cancel
G41		D	Cutter Compensation Left
G42		D	Cutter Compensation Right
G43		E	Tool Length Compensation (+)
G43.3		E	Tool Length Compensation (+) with Axis Tilt Compensation
G44		E	Tool Length Compensation (-)
G49	*	E	Tool Length Compensation Cancel
G50	*	M	Scaling/Mirroring Off
G51		M	Scaling/Mirroring On
G52		B	Offset Local Coordinate System Origin
G53		B	Rapid Position in Machine Coordinates
G54		L	Select Work Coordinate System #1
G55		L	Select Work Coordinate System #2
G56		L	Select Work Coordinate System #3
G57		L	Select Work Coordinate System #4
G58		L	Select Work Coordinate System #5
G59		L	Select Work Coordinate System #6
G61		F	Modal Decel and Stop *(formerly known as Exact Stop Mode)*
G64	*	F	Smoothing mode selection / Cancel Modal Decelerate and Stop
G65		J	Call Macro
G68		N	Coordinate Rotation on
G69	*	N	Coordinate Rotation off
G73		G	High Speed Peck Drilling
G74		G	Counter Tapping
G76		G	Fine Bore Cycle
G80	*	G	Canned Cycle Cancel

G81		G	Drilling and Spot Drilling
G82		G	Drill with Dwell
G83		G	Deep Hole Drilling
G84		G	Tapping
G85		G	Boring
G89		G	Boring with Dwell
G90	*	H	Absolute Positioning Mode
G91		H	Incremental positioning Mode
G92		B	Set Absolute position
G93		P	Inverse Time On
G93.1		P	Velocity Scrubber for Smoothed Inverse Time Data
G94		P	Inverse Time Off
G98	*	I	Initial Point Return
G99		I	R Point Return
G117		C	Rotation of Plane Selection XY
G118		C	Rotation of Plane Selection ZX
G119		C	Rotation of Plane Selection YZ
G173		G	Compound High Speed Peck Drilling
G174		G	Compound Counter Tapping
G176		G	Compound Fine Bore Cycle
G180		G	Compound Canned Cycle Cancel
G181		G	Compound Drilling and Spot Drilling
G182		G	Compound Drill with Dwell
G183		G	Compound Deep Hole Drilling
G184		G	Compound Tapping
G185		G	Compound Boring
G189		G	Compound Boring with Dwell

NOTES:

- All the default G-codes have been marked with the symbol " * ".

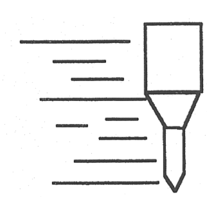
- A given line of a program may contain more than one G-code.

- If several G-codes from one group are used in the same line, only the G-code specified last will remain active.

- G-codes from group B are of "one shot" type (active only in the line in which they are specified). All other G-codes are modal (active until another G-code of the same group is specified).

- If a G-code from group A is used in a canned cycle mode, the canned cycle will be canceled. Canned cycle G-codes, however, have no effect on G-codes from group A.

G00 - Rapid Positioning



G0 moves to the specified position at the maximum motor rate. The coordinates may be either absolute positions (G90) or incremental positions (G91). G0 is modal and remains in effect until another positioning mode (G1, G2, G3 etc.) is commanded. G0 is the default-positioning mode.

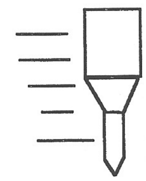
When the Z axis is commanded to move in the + direction, the Z axis will move up to its new position first, then the other axes will move to their new position along a straight line.

When the Z axis is commanded to move in the - direction, all axes but the Z axis will move to their new position along a straight line, then the Z axis will move down to its new position.

Example:
```
G0 X0.0 Y0.0 Z0.0      ; Rapid move to X0, Y0, Z0
```

CAUTION The feedrate override knob has no effect on G0 moves unless rapid override is turned ON

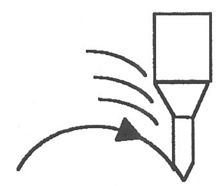
G01 - Linear Interpolation



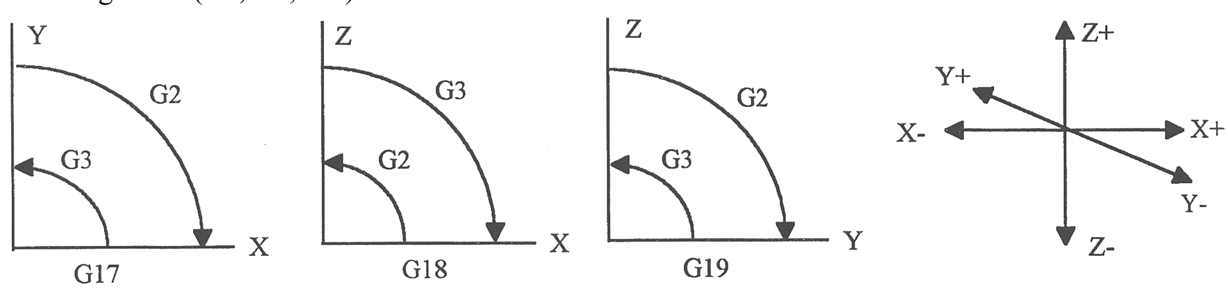
G1 moves to the specified position at the programmed feedrate. The coordinates may be either absolute positions (G90) or incremental positions (G91). The movement will be along a straight line. G1 is modal and remains in effect until another positioning mode (G0, G2, G3 etc.) is commanded.

Example:
```
G01 X2 Y3 Z4 W5 F10    ; Linear move to X2, Y3, Z4, W5 at a 10in/min
G91 X6 Y7              ; Linear move to X8, Y10
Z3 W4 F20              ; Linear move to Z7, W9 at 20in/min (G91 is modal)
```

G02 & G03 - Circular or Helical Interpolation



G2 moves in a clockwise circular motion, and G3 moves in a counterclockwise circular motion. This clockwise and counterclockwise motion is relative to your point of view, however. See the diagram below. The X, Y or Z position specified in the G2 or G3 command is the end position of the arc, and may be an absolute position (G90) or an incremental position (G91). G2 and G3 are modal and remain in effect until another positioning mode (G0, G1, etc.) is commanded.



View facing machine looking Z-

View facing machine looking Y+

View facing machine looking X-

*** NOTE:** When using G18, the G2 command moves in a counterclockwise direction in the XZ plane.

The axes included in the currently selected circular plane (G17, G18, or G19) will move in a circular motion. Any other axes specified will move along a straight line (helical movement). The programmed feedrate is used for the interpolated motion along the movement of all axes.

Helical and circular motion can be programmed in two different ways: specifying the final point and the radius of the arc, or specifying the final point and the parameters I, J, K (center point of the arc as incremental values from the start position).

* NOTE: For closed circles (arc of 360 degrees), use method 2: specify final point and parameters I, J and K. Method 1 (specify final point and radius) will not work.

METHOD 1: USING FINAL POINT AND RADIUS

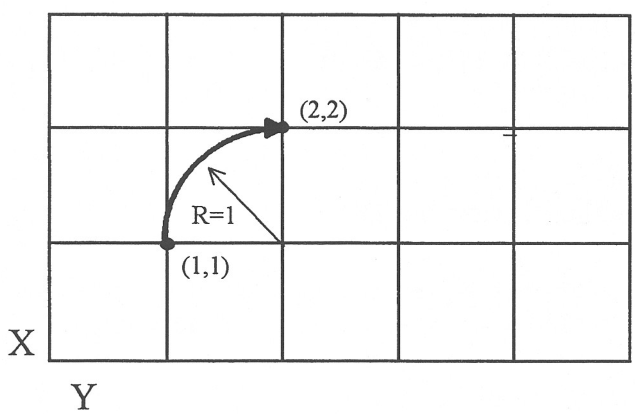
The commands G2 and G3 will have the following structure:

```
G2 Xa Yb Zc Rd
G3 Xa Yb Zc Rd
```

where *a*, *b*, and *c* will be the X, Y, and Z coordinates of the final point of the arc, and *d* will be the radius. In most cases there will be two possible arcs of the same radius connecting two given points. This occurs because the center of the arc is not specified. To choose the bigger arc, make the radius negative. To choose the smaller arc, make the radius positive. See examples 1 and 2 for graphical explanations of this concept.

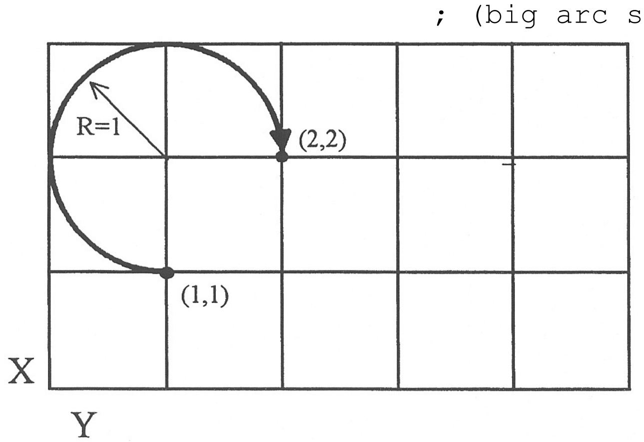
Example 1 (small arc solution: positive radius):

```
G17 G90 F25          ; selects XY plane and absolute positioning
G00 X1.0 Y1.0 Z0     ; rapid to start position X1, Y1, Z0
G02 X2 Y2 Z0 R1      ; arc to X2 Y2 Z0 with radius of 1
                     ; (small arc solution)
```



Example 2 (big arc solution: negative radius):

```
G17 G90 F25          ; selects XY plane and absolute positioning
G00 X1.0 Y1.0 Z0     ; rapid to start position X1, Y1, Z0
G02 X2 Y2 Z0 R -1    ; arc to X2 Y2 Z0 with radius of 1
                     ; (big arc solution)
```



METHOD 2: USING FINAL POINT AND PARAMETERS I, J, K

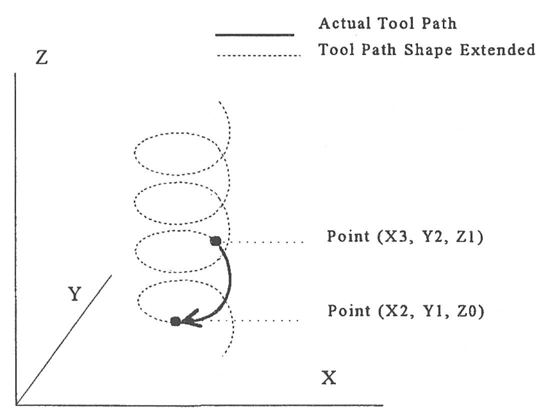
Another way to specify a helical or circular operation is using the parameters I, J, K instead of the radius R. The parameters I, J, and K are the **incremental** distances from the start point to the center of the arc. For absolute positioning on I, J, and K, parameter 2 bit 0 will need set. See the parameter section in Chapter 14.

I = X center - X start (valid for G17 & G18)
J = Y center - Y start (valid for G17 & G19)
K = Z center - Z start (valid for G18 & G19)

Examples:

Circular motion (See graph in method 1, example 2)

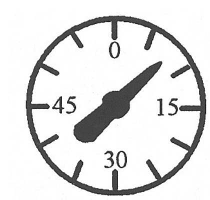
```
G17 G90 F25            ; selects XY plane and absolute positioning
G00 X1.0 Y1.0 Z0       ; rapid to start position X1, Y1, Z0
G02 X2 Y2 Z0 J1        ; arc to X2 Y2 Z0 with radius of 1
```



Helical motion

```
G17 G90 F30                      ; select XY plane and absolute positioning
G00 X3.0 Y2.0 Z1.0               ; rapid to start position X3, Y2, Z1
G02 X2.0 Y1.0 I-1.0 J0.0 Z0.0    ; CW XY arc from X3, Y2 to X2, Y1.
                                 ; Center at X2, Y2
                                 ; Helical Z move from 1 to 0
```

G04 - Dwell



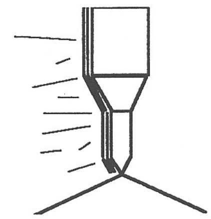
G4 causes motion to stop for the specified time. The P parameter is used to specify the time in seconds to delay. G4 causes the block to decelerate to a full stop.

The minimum delay is 0.01 seconds and the maximum is 327.67 seconds. The dwell time is performed after all motion is stopped and M functions on the line are completed. If the P parameter is not specified, X will be used instead. If neither P nor X is specified, the default dwell time of 0.01 seconds will be used.

Example:

```
G0 X1 Y1          ; rapid to X1, Y1
G4 P2.51          ; pause for 2.51 seconds
G1 X2 Y2          ; Linear move to X2, Y2
```

G09 - Decelerate and Stop *(formerly known as Exact Stop)*



G9 causes motion to decelerate to a stop and dwell for 1/100 seconds. G9 is equivalent to G4 P0.01. G9 is not modal; it is only effective for the block in which it appears. See G61 (Modal Decelerate and Stop).

Example:

```
G9 G0 X1 Y1       ; rapid to X1 Y1 and stop
X2 Y2             ; continue to X2 Y2
```

G10 - Parameter Setting

G10 allows you to set parameters for different program operations.

Examples:

```
G10 P73 R.05      ; Sets the peck drilling retract amount to .05
G10 P83 R.05      ; Sets the deep drill rapid down clearance to .05
G10 P81 R15       ; Sets G81 to use M15 instead of Z movement
G10 H5 R-1.3      ; Sets tool length offset #5 to -1.3 in the offset lib.
G10 D3 R.25       ; Sets tool diameter offset #3 to .25 in the offset lib.
```

G17, G18, G19 - Circular Interpolation Plane Selection

G17, G18, and G19 select the plane for circular interpolation commands (G02 & G03). G17 is the default plane. See figure under G2 and G3.

G17 is the XY plane
G18 is the ZX plane
G19 is the YZ plane

G20 - Select Inch Units

G20 selects inch units, affecting the interpretation of all subsequent dimensions and feedrates in the job file. G20 does not change the native machine units as set on the control setup menu.

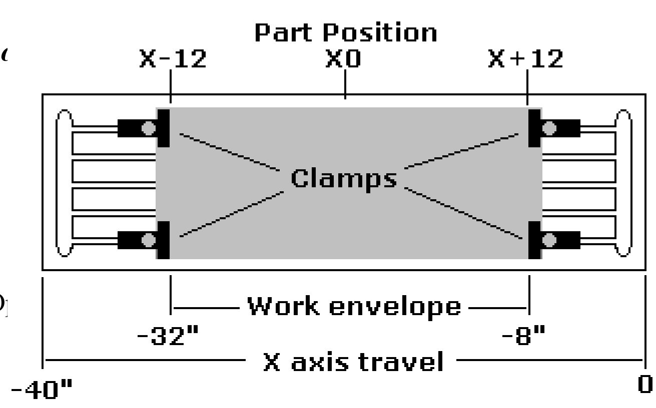
G21 - Select Metric Units

G21 selects metric units, affecting the interpretation of all subsequent dimensions and feedrates in the job file. G21 does not change the native machine units, as set on the control setup menu.

G22/G23 – Work Envelope On/Off

G22 turns on programmable work envelope in machine coordinates. When the machine tries to move into the forbidden area, let's say the x-axis, an "x-axis work envelope exceeded" message is displayed, letting you know which line of the program is at fault. The work envelope is set with the X, Y, Z for the '+' limit and I, J, K for the '–'limit. G22 is modal and remains on until turned off by G23 or the end of the job. The limits entered in the X, Y, Z and I, J, K parameters are stored in the WCS menu under **F3 - Work Envel**. For more information see chapter 4.

Example: (Machine h ***l in the X axis,)***




```
G22 X-8 I-32      ie outside 8 inches of
G1 X-13 F20       elope exceeded, line 3"
G23               rea.
```

```
M25             ; Z home
G0 X-13         ; Ok to move X here now
```

G28 - Return to Reference Point

G28 moves to the first reference point, by way of an intermediate point. The location of the reference point, in machine coordinates, may be set in Work Coordinate System Configuration. The intermediate point is specified in the local coordinate system, and may be at the current location (resulting in a move directly to the reference point). If an intermediate point is specified, only those axes for which positions are specified will be moved. If no axes are specified, all axes will be moved. The location of the intermediate point is stored for later use with G29.

Examples:

```
G28 G91 Z0              ; move Z-axis directly to reference point
                        ;( X and Y don't move)
G28 G91 X-.5 Y0 Z0      ; move X -0.5 (from current position), then move all
                        ; three axes to reference point
G28 G90 X2 Y4 Z.1       ; move all axes to (X2, Y4, Z0.1), then to
                        ; reference point
G28                     ; move all axes to the reference point
                        ;( no intermediate point)
```

* NOTE: As with G0 positioning moves, the Z-axis will move separately. If Z is moving up (the usual case) Z will move first, then the other axes. If Z is moving down, the other axes will move first, then Z. Because of this, it is rarely necessary to specify an intermediate point different from the current position.

G29 - Return from Reference Point

G29 moves all axes to the intermediate point stored in a preceding G28 or G30 command. It may be used to return to the work piece. If a position is specified, the machine will move to that position (in local coordinates) after reaching the intermediate point. G29 may only be specified after G28 or G30, though there may be intervening moves.

Examples:

```
G29                ; move all axes back from reference point to intermediate
                   ; point

G29 X1 Y2          ; move all axes to intermediate point, then move to X1 Y2
```

* NOTE: As with G0 positioning moves, the Z-axis will move separately. If Z is moving up, Z will move first, then the other axes. If Z is moving down (the usual case for G29), the other axes will move first, then Z will move.

G30 - Return to Secondary Reference Point

G30 functions exactly like G28, except that by default it uses the second reference point from the Work Coordinate System Configuration table, and the P parameter may be used to request either reference point.

Examples:

```
G30 G91 Z0         ; move Z axis directly to second reference point

G30 P1             ; move all axes to first reference point
```

NOTE: G30 P1 is equivalent to G28.

G40, G41, G42 -Cutter Compensation

G41 and G42 in conjunction with the selected tool diameter (D code) apply cutter compensation to the programmed tool path.

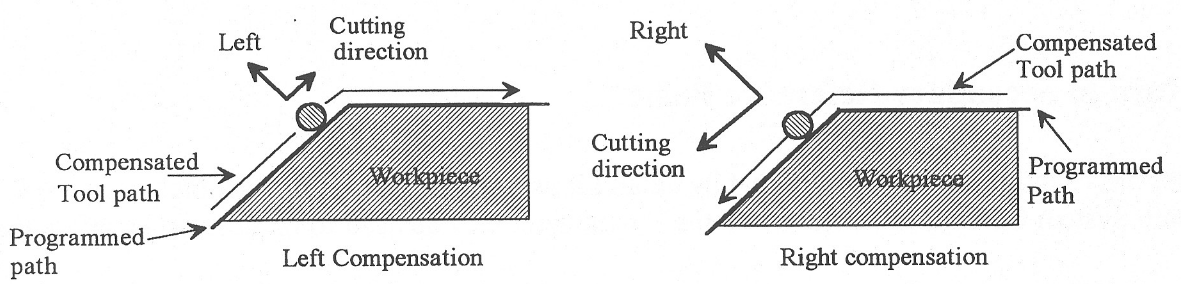
G41 offsets the cutter tool one half of the tool diameter selected with a D code, to the left of the work piece, relative to the direction of travel.

G42 offsets the cutter tool one half of the tool diameter selected with a D code, to the right of the work piece, relative to the direction of travel.

G40 cancels G41 and G42.

Example:

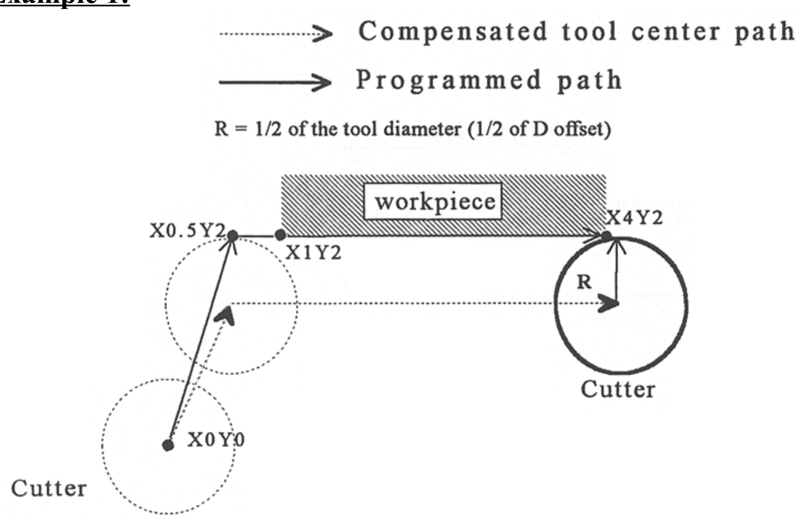
```
G41 D03            ; Tells the machine to compensate left half of the
                   ; diameter of the amount that corresponds to D03 in the
                   ; Tool Library
```



Whenever cutter compensation is applied, the following factors must be taken into account in order to obtain proper results.

1. The cutter diameter compensation function (G41, G42) must be implemented before the cutter tool reaches the starting cutting point.

<u>Example 1:</u>




```
G0X0Y0        ; Rapid tool to X0, Y0
G42 D3        ; Turn cutter compensation on, with a diameter of D3
G0X.5Y2       ; Rapid to X0.5, Y2
G1x4.1Y2      ; Linear cut to X4.1, Y2.
              ; Cut to X4.1 to clear material.
G40           ; Turn cutter compensation off.
```

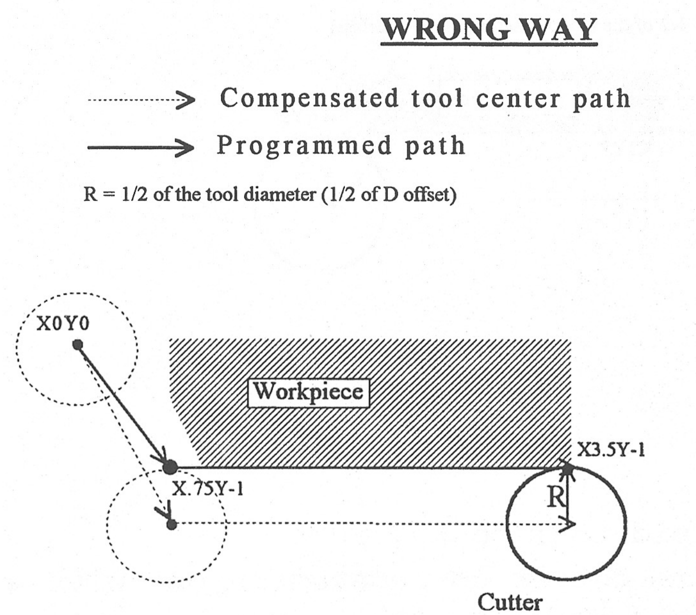
```
G0X5Y0      ; Rapid to X5, Y0.
```

You may want to add .1 or .05 inches on the final position for the last cut to clear the material.

NOTE: The diameter compensation statement G42 is placed before G0 X.5 Y2. As a result, the compensation is applied before the cutter reaches the starting cutting point X.5 Y2.

2. If the cutter is down, then the cutter compensation lead-in must always come from an appropriate direction. Otherwise, the work piece will be incorrectly cut, and the cutter tool could be damaged. One way to avoid this problem is by always keeping the cutter above the work piece whenever a transition is being made to a new starting cutting point. If for some reason this was not possible, then the G-code program should be written so that the cutter compensation lead-in paths do not interfere with the space occupied by the work piece. Example 2 illustrates a possible harmful outcome of programming an inappropriate lead-in direction.

Example 2:

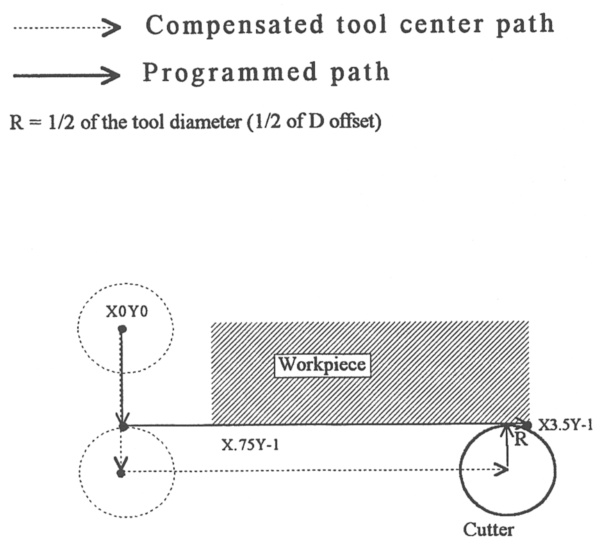



```
G0 X0Y0           ; Rapid tool to X0, Y0
G42 D5            ; Turn cutter compensation on, with a diameter of D5
G1 X.75Y-1 F5     ; Linear move to X0.75, Y-1. (Notice this damages the
                  ; corner of the work piece)
X3.6              ; move X to 3.6
G40               ; Turn cutter compensation off.
G0 X4Y-2          ; Rapid to X4, Y-2
```

* NOTE: This problem could have been avoided by selecting a transitional point between X0 Y0 and X.75 Y-1. A transitional point such as X-1 Y-1 would properly modify the lead-in path, keeping the cutter from damaging the corner of the work piece. Example 3 shows the correct way of performing this operation.

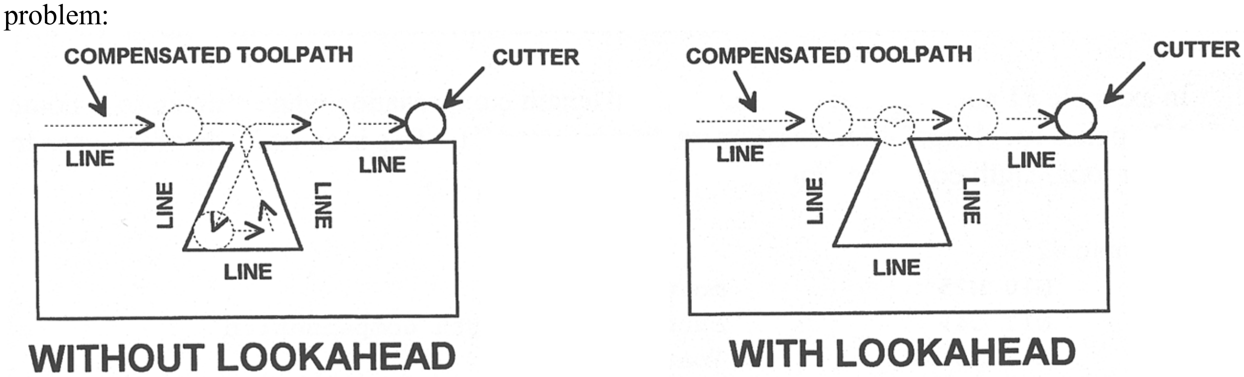
Example 3:

CORRECT WAY




```
G0X0Y0              ; Rapid tool to X0, Y0
G42D5               ; Turn cutter compensation on, with a diameter of D5
G0X0Y-1             ; Rapid tool to X0, Y-1
G1X.75Y-1           ; Linear cut to X0.75, Y-1.
X3.6                ; move X to 3.6
G40                 ; Turn cutter compensation off.
```

3. *Lookahead*. When the control machines any rapid traverse (G0), line (G1), or arc (G2, G3) with tool diameter compensation enabled, the program will look up to N consecutive events ahead of the current event in order to anticipate tool path clearance problems, where N is the number set in Parameter 99. Lookahead ensures that compensated tool paths don't overlap in programmed part sections where there is not enough clearance for the tool. The figure below shows a compensated tool path, and the actual tool path after Lookahead corrects the clearance problem:



Refer to the “Machine Parameters” section in Chapter 14 for more information on Parameter 99.

G43, G44, G49 - Tool Length Compensation

G43 and G44 apply tool length compensation to a selected tool to allow the control to utilize multiple tools in a single CNC program.

G43 applies positive compensation (from Z zero up). Work from part surface up.
G44 applies negative compensation (from Z zero down), used only when there is an absolute machine home. The spindle face is considered a zero length tool and all offsets are from there down.

G49 cancels tool length compensation (also canceled by issuing G43 H00).

Example:
```
G43 H01      ; tells the machine to offset the amount that
             ; corresponds to H01 in the Offset Library
```

G43.3 - Tool Length Compensation (+) with Axis Tilt Compensation

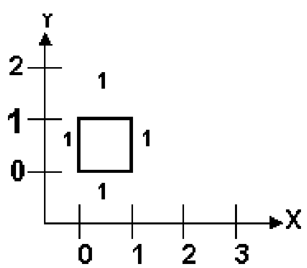
G43.3 is a special compensation mode which applies positive tool length compensation on a selected tool, just like G43, but also with additional X and Z compensations due to 5th axis tilt. This compensation mode is available only on those machines configured with a triangular rotary 5th axis (see parameter 166 in Chapter 14). Note that this compensation mode is the equivalent to G43 as long as the 5th axis is not tilted (i.e. local position is 0). G49 cancels this compensation mode.

G50, G51 - Scaling / Mirroring (Optional)

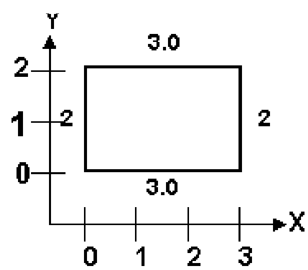
G50 and G51 scales program G-codes relative to a scaling center point defined as position (X, Y, Z).
A G51 applies scaling/mirror to all positions, lines, and arcs following this G-code, until a G50 is entered. Specify scaling factors with a value I, J, K. The X, Y, and Z parameters are the coordinates of the scaling center. If the scaling center is not specified, the default scaling center is the current cutter position as shown on the DRO. To mirror, enter a negative value for the scaling factor.

Example, Scaling:

```
G51 X0.0 Y0.0 Z0.0 I3.0 J2 K1     ; turn scaling on
G00 X0.0 Y0.0 Z1.0                ; rapid to X0, Y0, Z1
G01 X1.0 Y0.0 Z1.0                ; line to X1, Y0, Z1
G01 X1.0 Y1.0 Z1.0                ; line to X1, Y1, Z1
G01 X0.0 Y1.0 Z1.0                ; line to X0, Y1, Z1
G01 X0.0 Y0.0 Z1.0                ; line to X0, Y0, Z1
G01 X0.0 Y0.0 Z0.0                ; line to X0, Y0, Z0
G50                               ; cancel scale
```



Original Box

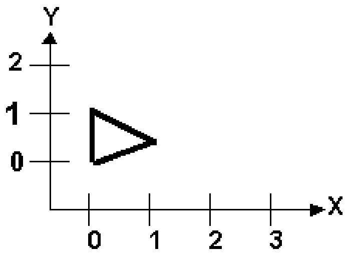


Scaled Box

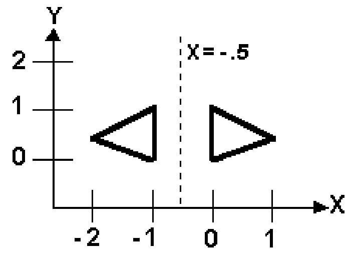
For this G51, the following program lines were scaled 3:1 in the X direction, 2:1 in the Y direction, and 1:1 in the Z direction. If no scale factor is specified, the default is 1:1 for all axes.

Example, Mirroring:

```
G51 X-0.5 Y0.0 Z.0 I-1 J1 K1      ; turn mirror on (x axis -0.5 mirror
                                  ; line)
G00 X0.0 Y0.0 Z1.0                ; rapid traverse to X0, Y0, Z1
G01 X1.0 Y0.5 Z1.0                ; line to X1, Y.5, Z1
G01 X0.0 Y1.0 Z1.0                ; line to X0, Y1, Z1
G01 X0.0 Y0.0 Z1.0                ; line to X0, Y0, Z1
G50                               ; cancel mirror
```



Original Triangle



Mirrored Triangle

If scaling factors are the same for all the axes, parameter P can be used.

Example:
```
G51 X1.0 Y2.0 Z0.0 P2.5          ; scale all axes a factor of 2.5.
```

If an arc is scaled with uneven scaling factors, the result will depend on how the arc center and radius were specified:
1. If the arc radius was specified with R, the radius will be scaled by the larger of the two circular plane scale factors. The result will be a circular arc between the scaled arc start and the scaled arc end.
2. If the arc center was specified with I, J, and/or K, the centers will be scaled by the appropriate axis scale factors. The result will be a circular arc from the scaled arc start, around the scaled center, and usually with a line from the end of the circular arc to the scaled arc end.
3. In no case can an ellipse be generated using scaling.

G52 - Offset Local Coordinate System

G52 shifts the local coordinate system origin by a specified distance. Multiple G52 codes are not cumulative; subsequent shifts replace earlier ones. The G52 shift may therefore be canceled by specifying a shift of zero. If you are using multiple coordinate systems, the G52 shift amount will affect all coordinate systems.

Example:

```
G0 X0 Y0          ; move to origin
M98 P9100         ; call subprogram
G52 Y4            ; shift coordinate system 4 inches in Y
G0 X0 Y0          ; move to new origin
M98 P9100         ; call subprogram again with new coordinates
G52 Y0            ; restore unshifted coordinate system
```

G53 - Rapid Positioning in Machine Coordinates

G53 is a one shot code that performs a rapid traverse using machine coordinates. It does not affect the current movement mode (G0-G3) or coordinate system (G54-G59). G53 may only be used with absolute positioning (G90).

Example:

```
G53 X15 Y4 Z0     ; move to 15,4,0 in machine coordinates
```

G54 - G59 - Select Work Coordinate System

G54 through G59 select among the six regular work coordinate systems (WCS #1 through WCS #6). After issuing the code, subsequent absolute positions will be interpreted in the new coordinate system. Alternatively, the codes E1 through E6 to can be used instead of G54 through G59.

Example:

```
G54 G0 X0 Y0 Z0        ; select first WCS, move to origin
G2 X1 I.5 Z-.5         ; mill something...
G0 Z.1                 ; Rapid to position Z0.1
G55 X1 Y1              ; select second WCS, move to X1, Y1
```

Using Extended Work Coordinate Systems (optional): There are 12 additional work coordinate systems available as an extra-cost option. In a G-code program, these 12 additional work piece origins may be selected with either “G54 P1” (WCS # 7) through “G54 P12” (WCS #18) or “E7” through “E18.”

Regular WCS

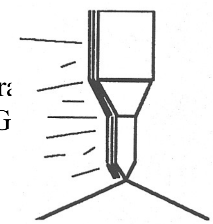
WCS	G-Code	E Code
WCS #1	G54	E1
WCS #2	G55	E2
WCS #3	G56	E3
WCS #4	G57	E4
WCS #5	G58	E5
WCS #6	G59	E6

Extended Work Coordinate Systems (optional)

WCS	G-Code	E Code
WCS #7	G54 P1	E7
WCS #8	G54 P2	E8
WCS #9	G54 P3	E9
WCS #10	G54 P4	E10
WCS #11	G54 P5	E11
WCS #12	G54 P6	E12

WCS	G-Code	E Code
WCS #13	G54 P7	E13
WCS #14	G54 P8	E14
WCS #15	G54 P9	E15
WCS #16	G54 P10	E16
WCS #17	G54 P11	E17
WCS #18	G54 P12	E18

G61 - Modal Decelerate and Stop
(formerly known as Exact Stop Mode)



G61 activates Decelerate and Stop mode for every block processed. This forces motion to decelera invokes a brief dwell (1/100 seconds) at the end of each block (equivalent to G9 in each block). G remains in effect until it is canceled with G64. Note that G61 also turns off Smoothing mode.

Example:

```
G0 X0 Y0           ; move to origin
G61 X2             ; move and decelerate and stop at X2
X4                 ; move and decelerate and stop at X4
X5                 ; move and decelerate and stop at X5
```

G64 – Smoothing Mode Selection / Cancel Modal Decel and Stop

G64 has multiple formats with different functionality. Invoking G64 with either an ON or OFF parameter sets the Smoothing mode to on or off, and also cancels Modal Decelerate and Stop (G61). Invoking G64 without either the ON or OFF parameter simply cancels Modal Decelerate and Stop. Note that all forms of G64 will cancel Modal Decelerate and Stop (Cancel G61).

G64 Format	Function
`G64`	Simply cancel Modal Decelerate and Stop (Cancel G61). This does not affect Smoothing mode.
`G64 ON`	Turn on Smoothing mode using the current parameter settings (in P221-P228, P230, and P231). The effect of this command is temporary as it is active only until the next G64 ON/OFF command or until the end of job. Upon the start of the next job, the the inital on/off state of Smoothing will once again be determined by parameter 220. This command does not modify any parameters. (This also cancels G61.)
`G64 OFF`	Turn off Smoothing mode. The effect of this command is temporary as it is active only until the next G64 ON/OFF command or until the end of job. Upon the start of the next job, the the inital on/off state of Smoothing will once again be determined by parameter 220. This command does not modify any parameters. (This also cancels G61.)
`G64 ON "preset label"` `G64 ON P__`	Turn on and activate Smoothing mode using the specified “preset label” or preset number (P1-P99). (See Chapter 14 for further information on setting up Smoothing Presets.) Note that “preset label” is not case

(where P__ is P1-P99)	sensitive. You can use the preset number to specify those presets that don't have a "preset label". These commands have a lasting effect beyond the end of the current job because they actually copy the specified Smoothing preset settings into the current parameters (P221-P228, P230, and P231) and sets P220 to 1. (This also cancels G61.)
`G64 ON P0` `G64 OFF P0`	Turn off and deactivate Smoothing mode. Both forms of this command do the same thing. The P0 refers to "Exact Stop" mode. This command sets P220 to 0, and thus has a lasting effect beyond the end of the current job. (This also cancels G61.)

Examples:

```
G64                      ;cancel Modal Decelerate and Stop
G64 ON                   ;turn on modal Smoothing mode
G64 OFF                  ;turn off modal Smoothing mode
G64 ON P2                ;Activate Smoothing Preset #2 by number
G64 ON "contouring mill" ;Activate "Contouring Mill" Preset by label
G64 ON P0                ;Deactivate Smoothing. Activate Exact Stop mode.
```

G65 - Call Macro

G65 calls a macro with user-specified values. A macro is a subprogram that executes a certain operation (e.g. drill pattern, contours, etc.) with values assigned to variable parameters within the operation.

<u>**Calling methods:**</u>

G65 Pxxxx Lrrrr Arguments
or
G65 "program.cnc" L*rrrr Arguments*

where *xxxx* is the macro number (referring to file O*xxxx*.cnc, 0000-9999 allowed, leading zeros required in filename, capital O, lowercase .cnc), *rrrr* is the repeat value, "program.cnc" is the name of the macro file, and *Arguments* is a list of variable identifiers and values.

Arguments to macro calls are specified by using letters A-Z, excluding G, L, N, O, and P.

Macros are written just like normal programs. However, macro programs may access their arguments by using #A, #B, etc., or by using numbers: #1 for A, #2 for B, etc. (exceptions: #4-6 for I-K, #7-11 for D-H). Arguments I, J, and K can be used more than once in a macro call, with the first set of values stored as #4-6, the second as #7-9, etc., to a maximum of 10 sets. See example at the end of this G65 section.

Macros 9100 - 9999 may be embedded into a main program, using O91*xx* to designate the beginning of the macro and M99 to end it. The CNC software will read the macro and generate a file O91*xx*.cnc, but will not execute the macro. It will be executed when G65 is issued.

Example 1:

Main program:

```
G65 "TEST.cnc" A5 B3 X4
```

Macro TEST.cnc:

```
G1 X#X Y#A Z-#B
```

This call will produce

```
G1 X4 Y5 Z-3
```

Example 2:

Main program:

```
G65 "TEST2.cnc" I5 J3 K40 I-1 J2 I0 J0
```

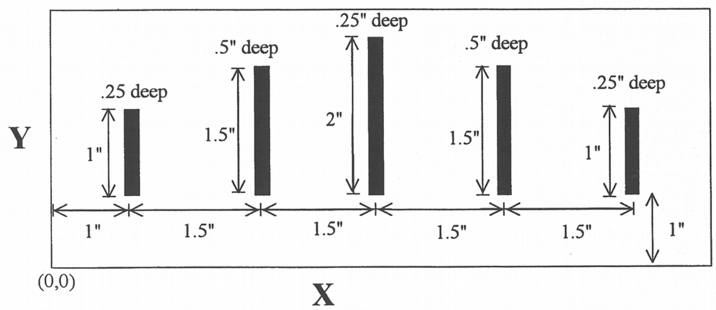
Macro TEST2.cnc:

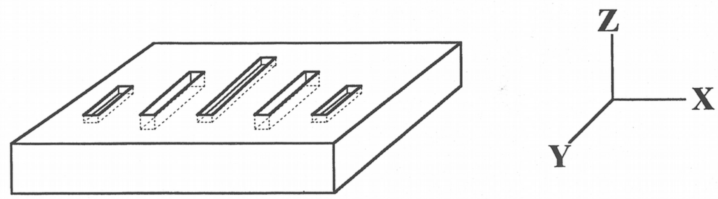
```
G1 X#4 Y#5 F#6
G1 X#7 Y#8 Z#9
G1 X#10 Y#11 Z#12
```

This call will produce

```
G1 X5 Y3 F40
G1 X-1 Y2 Z0
G1 X0 Y0 Z0
```

Example 3:
Suppose a piece is to have notches of different lengths and depths along the x-axis:





The macro variables would handle the length in the Y direction and depth in the Z direction:

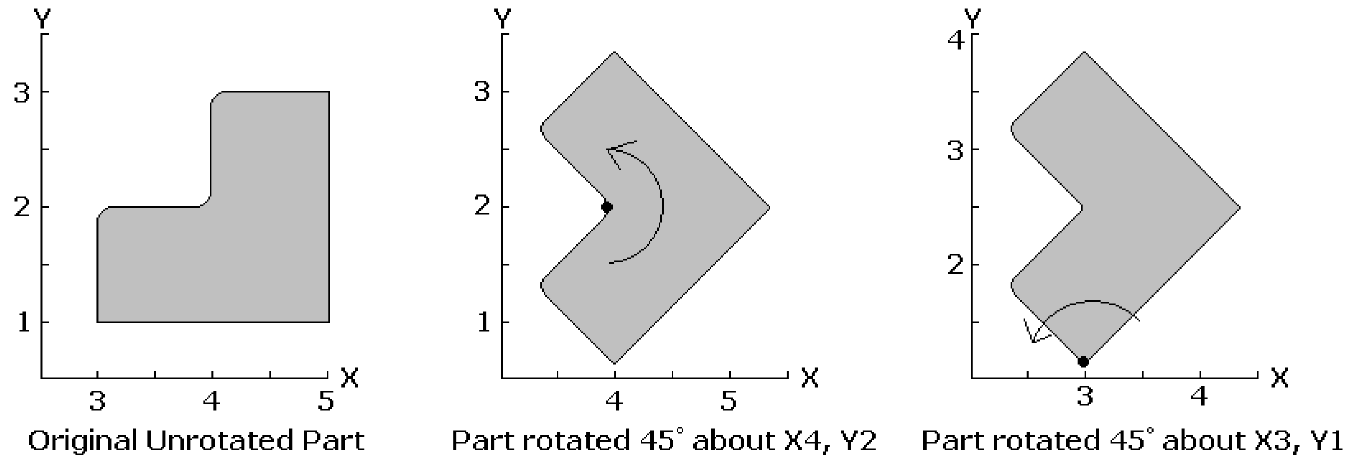
```
O0002
G90 G1 Z0 F30          ; Linear move to Z0
Z#Z F5                 ; Cut to variable depth
G91Y#Y F10             ; Cut variable length
G90 G0 Z0.1            ; Retract
```

The main program would call this macro five times, each time specifying the depth and length required.

```
: Main Program
G90 G0 X1 Y1 Z0.1          ; Move to first notch
G65 P0002 L1 Y1 Z.25       ; Call macro and assign Y=1" and Z=.25"
G90 G0 X2.5 Y1             ; Move to second notch
G65 P0002 L1 Y1.5 Z.5      ; Call macro and assign Y=1.5" and Z=.5"
G90 G0 X4 Y1               ; Move to third notch
G65 P0002 L1 Y2 Z.25       ; Call macro again
G90 G0 X5.5 Y1             ; Move to fourth notch
G65 P0002 L1 Y1.5 Z.5      ; Call macro again
G90 G0 X7 Y1               ; Move to fifth notch
G65 P0002 L1 Y1 Z.25       ; Call macro again
: End program
```

G68, G69 - Coordinate Rotation on/off

G68 rotates program G-codes a specified angle R. G68 rotates all positions, lines, and arcs until a G69 is entered. The center of rotation can be specified by X, Y and Z values (X, Y for G17 plane). If the center is not specified then a default center of rotation is used as determined by machine parameter #2 (see Chapter 14 for parameter #2). The default plane of rotation is G17 (X, Y).



Example:

```
G68 R45 X4 Y2                ; Rotate 45 degrees centered on X4 Y2
G0 X3.0 Y1.0                 ; Rapid to position
G1 X5.0 Y1.0 F20             ; Start part profile
X5.0 Y3.0
X4.125 Y3.0
G3 X4.0 Y2.875 J-0.125
G1 X4.0 Y2.125
G2 X3.875 Y2.0 I-0.125
G1 X3.125 Y2.0
G3 X3.0 Y1.875 J-0.125
G1 X3.0 Y1.0                 ; End part profile
G69                          ; Rotate Off
```

G73, G76, G80, G81, G82, G83, G85, G89 - Canned Drilling/Boring Cycles

G74, G84 - Canned Tapping Cycles

G code	-Z direction (machine hole)	Operation at bottom of hole	+Z direction	Use
G73	Intermittent Feed (Set with the Q parameter)	----------	Rapid traverse	High speed peck drilling cycle
G74	Feed	Spindle CW, then Dwell (Set with the P parameter)	Feed	Counter tapping (Left-hand thread)
G76	Feed	Dwell (P parameter), Orient Spindle (via M19), Move Y+ (Q parameter)	Rapid traverse, then Stop Spindle Orient (via M5)	Fine Boring Cycle
G80	----------	----------	----------	Cancels canned cycles
G81	Feed	----------	Rapid traverse	Regular and spot drilling cycles and air drill cycle

G82	Feed	Dwell (Set with the P parameter)	Rapid traverse	Regular and counter boring cycles, spot facing
G83	Intermittent Feed (Set with the Q parameter)	----------	Rapid traverse	Peck and deep hole drilling cycles
G84	Feed	Spindle CCW, then Dwell (Set with the P parameter)	Feed	Tapping (Right-hand thread)
G85	Feed	----------	Feed	Boring cycle
G89	Feed	Dwell (Set with the P parameter)	Feed	Boring cycle

Table 1. Canned drilling, boring and tapping cycles

Canned Cycle Operation

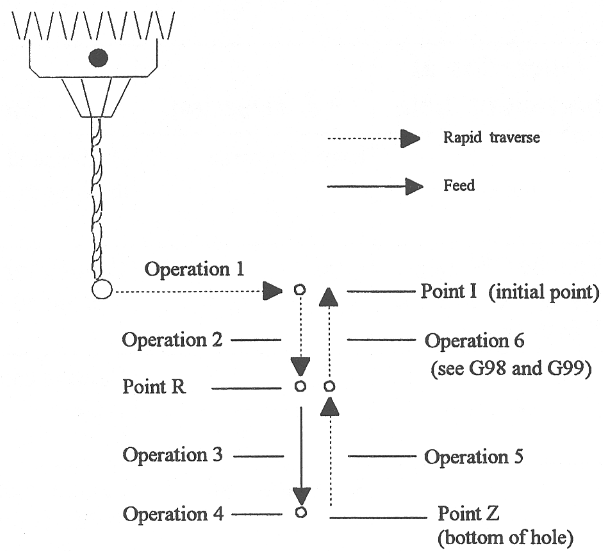
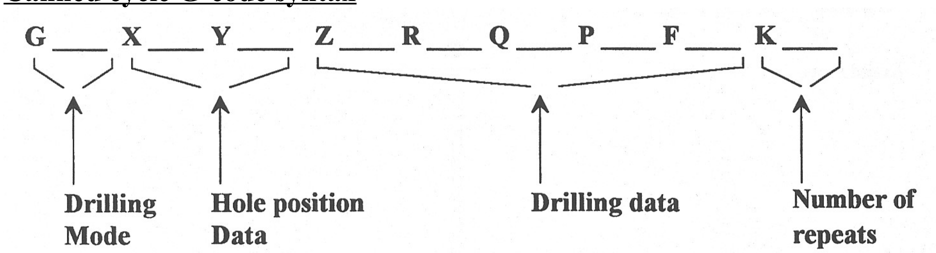


Figure 1. Drilling cycle operation

Operation 1: Position the X, Y axes.
Operation 2: Rapid traverse to the position labeled R.
Operation 3: Machine hole.
Operation 4: Bottom hole operation.
Operation 5: Return to point R.
Operation 6: Rapid traverse to initial point.

Canned cycle G-code syntax



(Cycle codes do not have to be on the same line)

G ____ Canned cycle G-code from table 1.

X ____ X position of the hole to be drilled.

Y ____ Y position of the hole to be drilled.

Z ____ Specifies point Z in figure 1.
In incremental mode Z is measured from point R.
In absolute mode Z is the position of the hole bottom.

R ____ Specifies the distance to point R (figure 1) with an absolute or incremental value.

Q ____ Determines the cut-in depth for the G73 and G83 cycles.
Determines the thread lead for G74 and G84 if Rigid Tapping is enabled.
(In the case of Rigid Tapping Q is not modal)

P ____ Sets the dwell time at the bottom of the holes for G74, G82, G84, and G89 cycles. The dwell time is measured in seconds (same as G04).

F ____ Sets the feed rate. Remains the feedrate even after G80 (cancel canned cycles).

K ____ Sets the number of repeats for drilling cycles. Operations 1 through 6 of figure 1 will be repeated K number of times. If K is not specified K = 1. K is only useful when using incremental positioning mode (G91) and is not retained from cycle to cycle. In absolute mode, K causes the drilling of the same hole in the same position K times.

* NOTE: Canned cycles are modal and should be canceled with G80. However G00, G01, G02 and G03 will also cause the cancellation of canned cycles. All parameters are stored until canned cycles are canceled except for the hole position and K, which must be set each time the cycle is used. When G80 is issued the movement mode will be the last one issued (G0, G1, G2, G3). Canned cycles will not be performed unless X and/or Y are specified.

When performing canned cycle operations, the distances can be either incremental or absolute, depending on the current active mode (G90 = absolute, G91= incremental). Figure 2 illustrates canned cycle Z-axis distances in both modes.

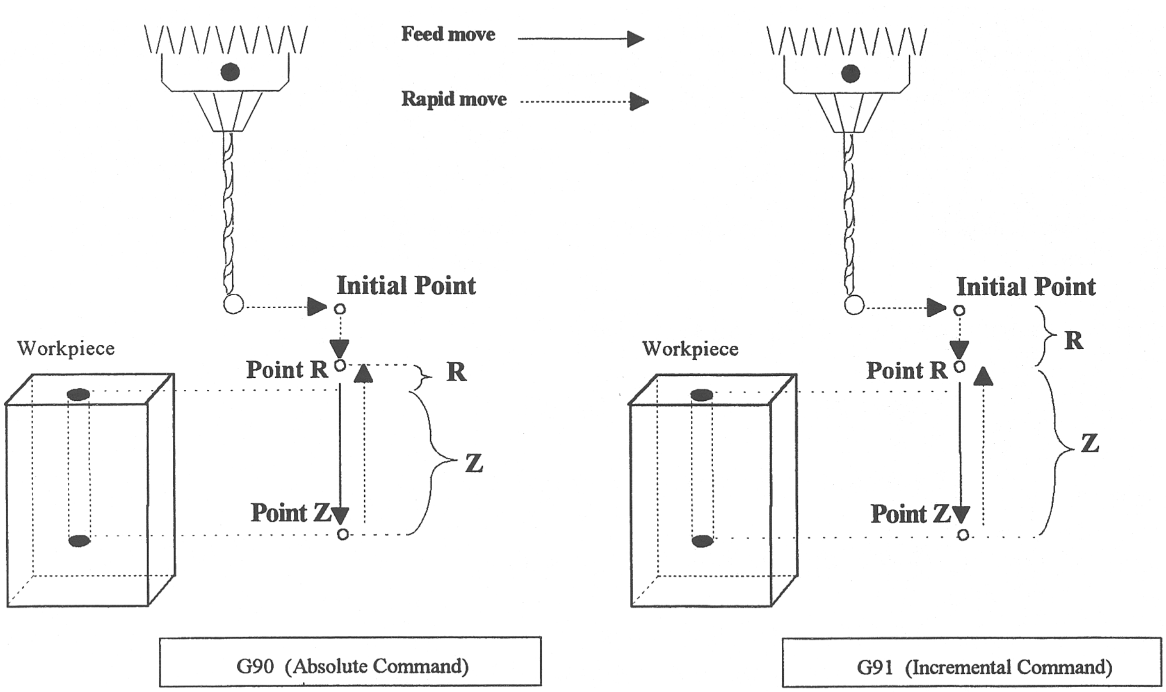


Figure 2: Canned Cycle Absolute and Incremental modes

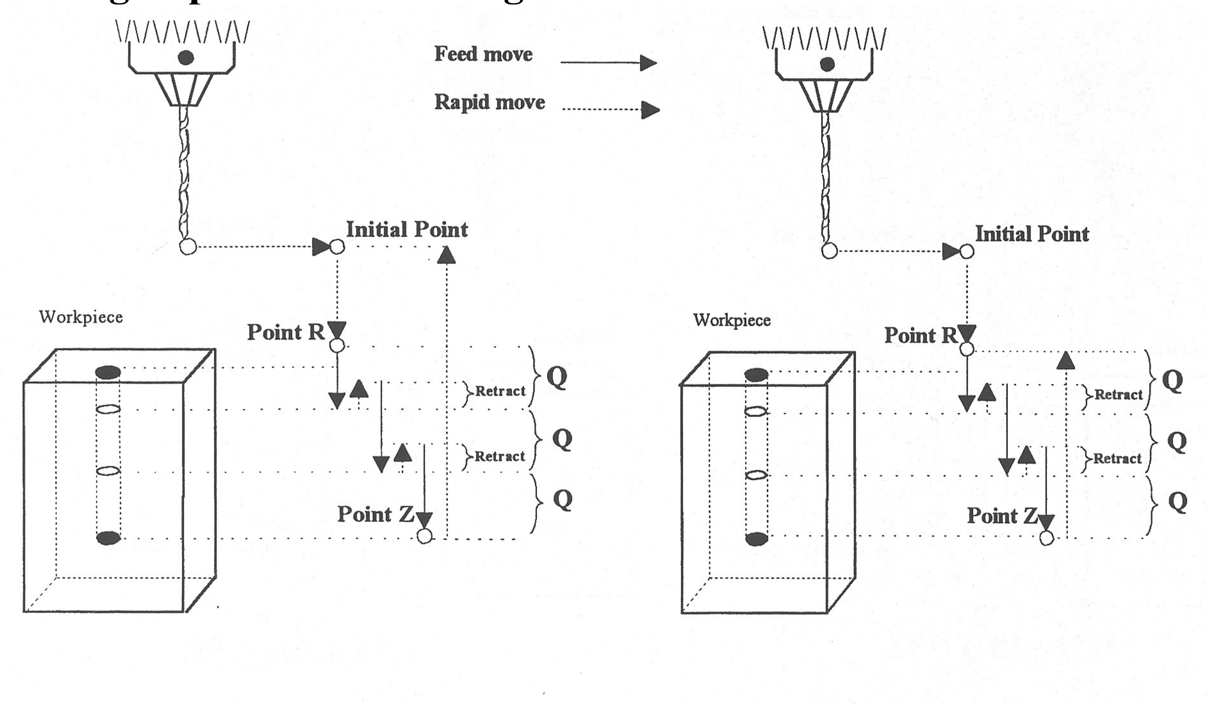
* NOTE: In incremental mode the Z depth of the hole is measured from R, and R is measured from the initial tool position.

Example:
(Part surface height is Z = 0, initial tool position is X.50 Y1.0 Z.625. Drill 0.50 deep hole at X1.0 Y1.0; clearance height (R) is 0.10 above surface.)

Absolute	Incremental
G90	G91
G81 X1 Y1 R.1 Z-.5	G81 X.5 Y0 R-.525 Z-.6
G80	G80

* NOTE for Articulated Head machines configured with the TWCS feature enabled via Parameter 166:
If the currently selected WCS is non-TWCS (TWCS = No) and the B axis is at an angle other than 0, then you cannot use the regular Canned Cycle G-codes G73, G74, G76, G81, G82, G83, G84, G85, G89. You must use the Compound Canned Cycle G-codes G173, G174, G176, G181, G182, G183, G184, G185, G189 instead. Using regular Canned Cycle G-codes when the B axis is not 0 is an error and will cancel the job. See "G173, G174... – Compound Canned Cycles" later in this chapter for more information about this subject. See Chapter 14 for more information about Parameter 166.

G73 - High Speed Peck Drilling



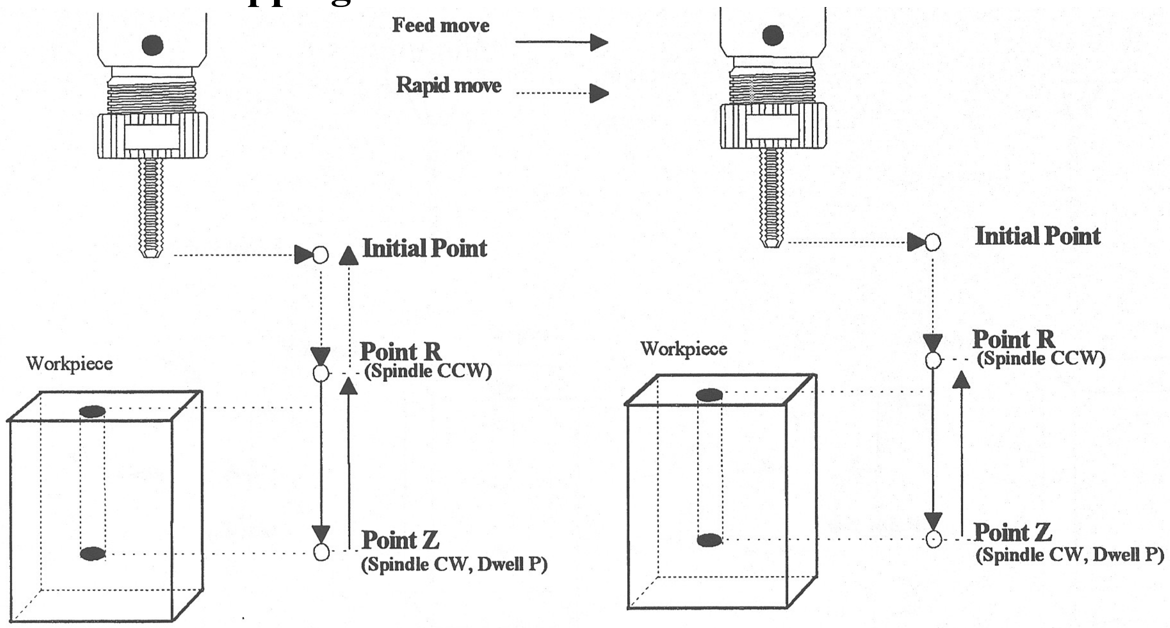
G73 using G98 **G73 using G99**

G73 is the peck drilling cycle. The hole is drilled in a series of moves: down a distance Q at a given feedrate, up the retract distance at the rapid rate, and then down again at the given feedrate. The retract amount is set with G10 as shown in the example below.

Example:
```
G90                                    ; Absolute positioning
G01 X3.00 Y1.50 Z.5                    ; G01 mode before canned cycle
G98                                    ; Set for initial point return
G10 P73 R.1                            ; Sets the retract amount to .1
G73 X3.250 Y1.75 Z-.650 R.1 Q0.325 F3  ; Peck drill at X3.25 Y1.75
X4.5 Y3.5                              ; Peck drill at X4.5 Y3.5
G80                                    ; Cancel canned cycle, return
```

```
                                        ; to G01
```

G74 - Counter Tapping



G74 using G98 **G74 using G99**

G74 performs left-hand tapping. The spindle speed (and feedrate, if you are doing floating tapping) should be set and the spindle started in the CCW direction before issuing G74. G74 will normally use the default M3 to select spindle CW (at the bottom of the hole) and M4 to re-select spindle CCW (after backing out of the hole) depending on the settings of parameters 74 and 84.

The tap may continue to cut a short distance beyond the programmed Z height as the spindle comes to a stop before reversing. When tapping blind holes, be sure to specify a Z height slightly above the bottom of the hole to prevent the tool from reaching bottom before the spindle stops.

Note: If rigid tapping is enabled, a Q may be used to set the thread lead or pitch. However, because Q is <u>not</u> modal in the case of Rigid Tapping, you must specify Q on every line at which Rigid Tapping is to occur.

Note: At the bottom of the hole, G74 will call the default version of the specified M function even if it has been customized by an M function macro.

WARNING FEED HOLD is temporarily disabled during the tapping cycle, but it will be re-enabled at the end of the cycle.

NOTICE Pressing CYCLE CANCEL while the tap is in the hole will very probably break the tap or strip the threads in the tap hole. However, do so if it is an emergency.

Example:

```
M4 S500 F27.78              ; start spindle CCW, set up for 18 pitch tap
G74 X1 Y1 R.1 Z-.5          ; counter-tap a 0.5 deep hole at X1 Y1
Y1.5                        ; ... and another one at X1 Y1.5
G80                         ; cancel canned cycles
```

G76 – Fine Bore Cycle

WARNING!!! G76 requires that the machine be capable of orienting the spindle and that a custom M19 macro is present in order to command the inverter to orient the spindle. Please contact your dealer to confirm that your machine meets these requirements before attempting to use this cycle.

NOTE: Parameter 136 = Fine Bore retract angle (0-360 degrees). A setting of 0 = Retract in Y+ direction

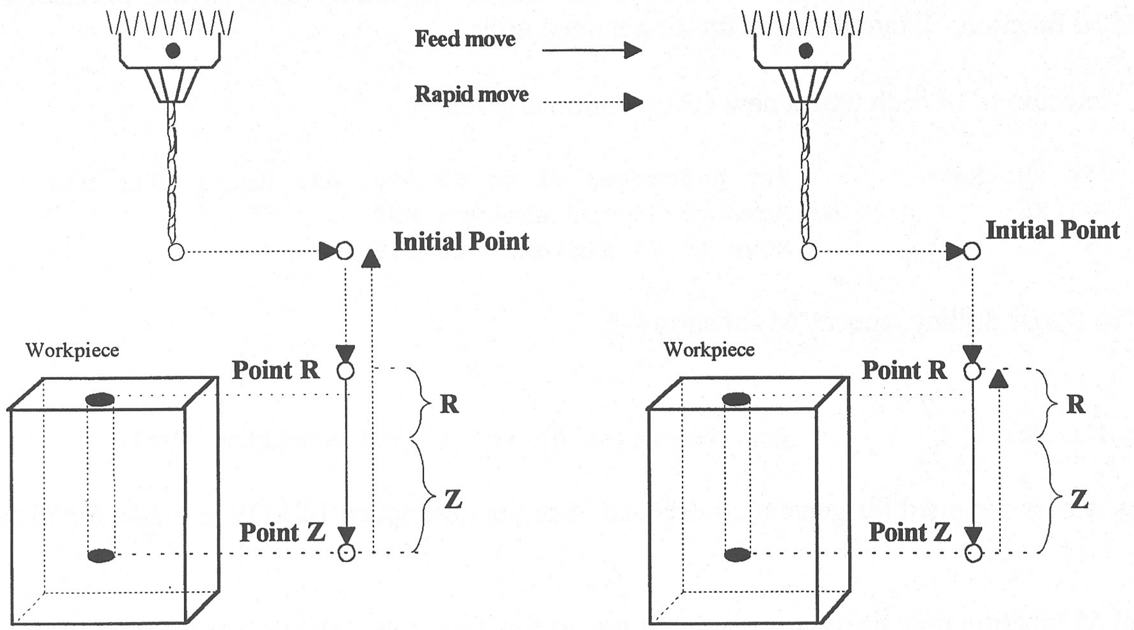
Format G76 X__ Y__ Z__ R__ Q__

R = Point R Q = Distance to pull away from wall in Y+ direction at bottom of hole.

Example :

```
G76 X1 Y1 Z-3 R.1 Q.2       ; Bore hole at X1 Y1 retract .2 in Y+ direction
Y10                         ; ... and another one at X1 Y10
G80                         ; cancel canned cycles
```

G81 - Drilling and Spot Drilling



G81 using G98 **G81 using G99**

G81 is a general purpose drilling cycle. The hole is drilled in a single feedrate move, and then the tool is retracted at the rapid rate.

Example:

```
G90                              ; Absolute positioning
G01 X3.00 Y1.50 Z.5              ; G01 mode before canned cycle
G99                              ; Set for R point return
G81 X3.250 Y1.75 Z-.650 R.1 F3   ; Drill at X3.25 Y1.75
X4.5 Y3.5                        ; Drill at X4.5 Y3.5
G80                              ; Cancel canned cycle, return to G1
```

G81 - Drill Cycle Transformation to G81 Air Drill Cycle

G81 may be modified to execute an M function instead of moving the Z-axis by setting parameter #81 to the desired M function. Example use is for air-actuated drills.

Example: Execute M39 each time a new G81 position is given:

```
G10 P81 R39            ; Set parameter 81 to 39 (G81 air drill with M39)
G81 X5                 ; Move to X5 and execute M39
Y3                     ; Move to Y3 and execute M39
```

To revert to Z-axis drilling, specify M function #-1.

Example:

```
G10 P81 R-1            ; Set parameter 81 to -1 (G81 drilling cycle)
```

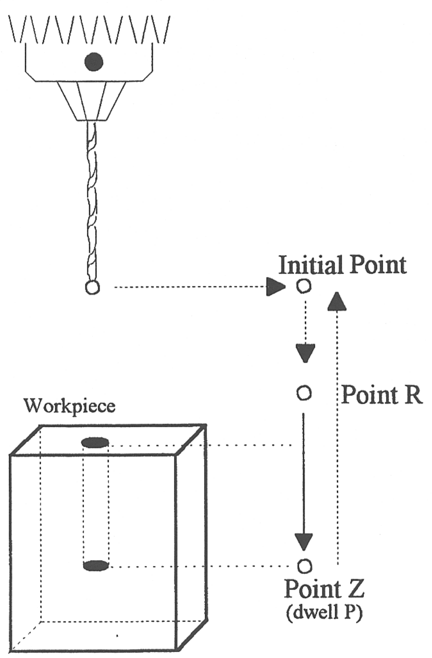
M function #39 is designed for general air drill use. See the description of M39 in the M functions section.

A different M function may be used instead, but any M function used must be a macro file that uses the M103 and M104 commands to time the cycle (see the example in the M function section under M103). If the macro file does not use M103, the control will automatically cancel the job 1/2 second after starting G81. For information on creating customized M functions, review Macro M functions in Chapter 13.

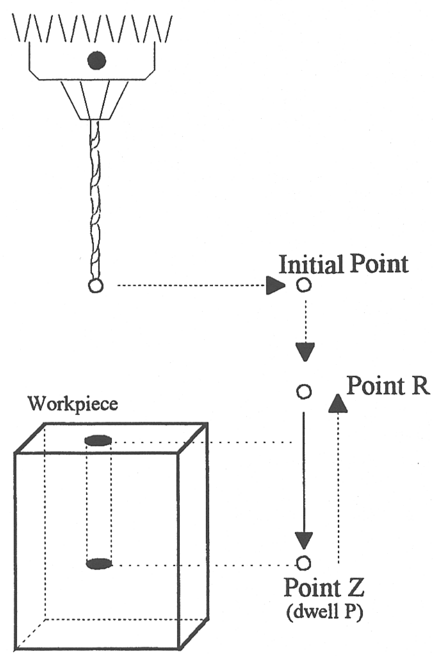
The M39 default air drill cycle has a time out of 2.0 seconds. As a result, if the cycle does not complete within 2 seconds then the cycle aborts and the output relay is turned off under PLC program control.

* NOTE: The PLC program must be involved in the execution of the cycle. The PLC program is responsible for turning on relays based on M function requests and the status of program execution. The PLC program must also stop all programmed machine functions when the program is canceled. See the M39 description (Chapter 13) for a sample of an air drill cycle M function.

G82 - Drill with dwell



G82 Using G98



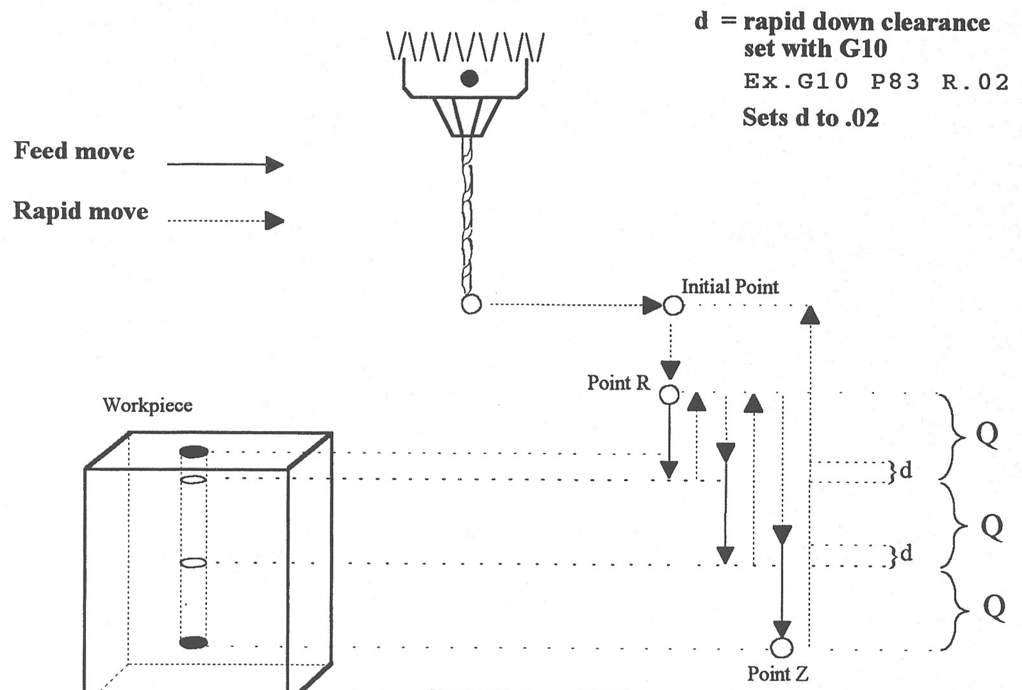
G82 Using G99

G82 is a general purpose drilling cycle similar to G81. However, G82 includes an optional dwell at the bottom of the hole before retracting the tool. This can make the depth of blind holes more accurate.

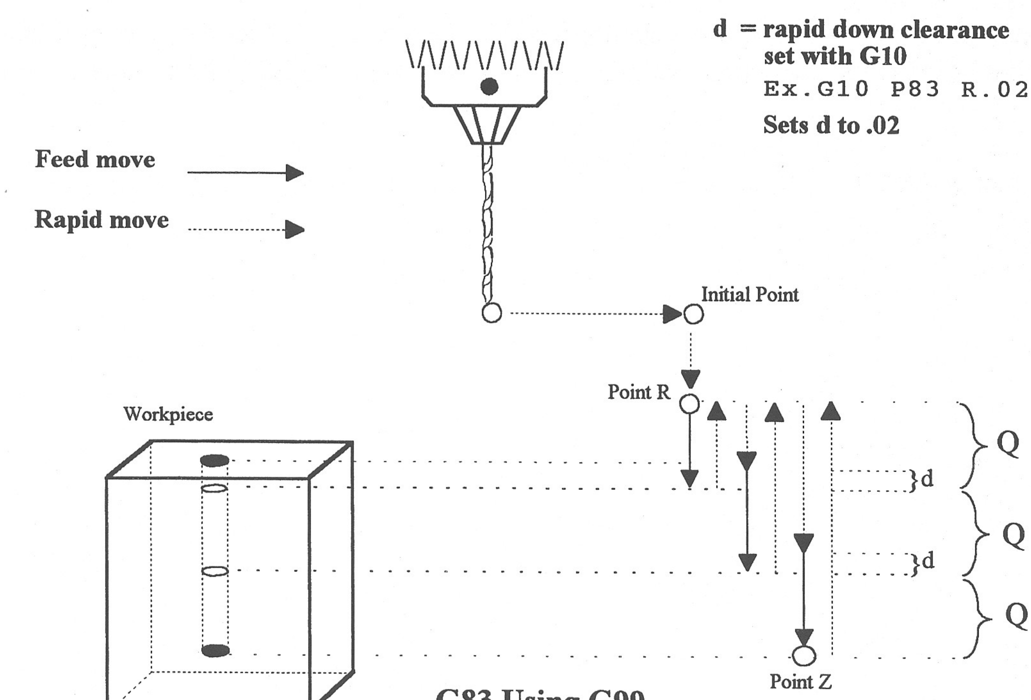
Example:

```
G82 X1 Y1 R.1 Z-.5 P.5                 ; drill to Z-.5, dwell .5 seconds
```

G83 - Deep hole drilling



G83 Using G98



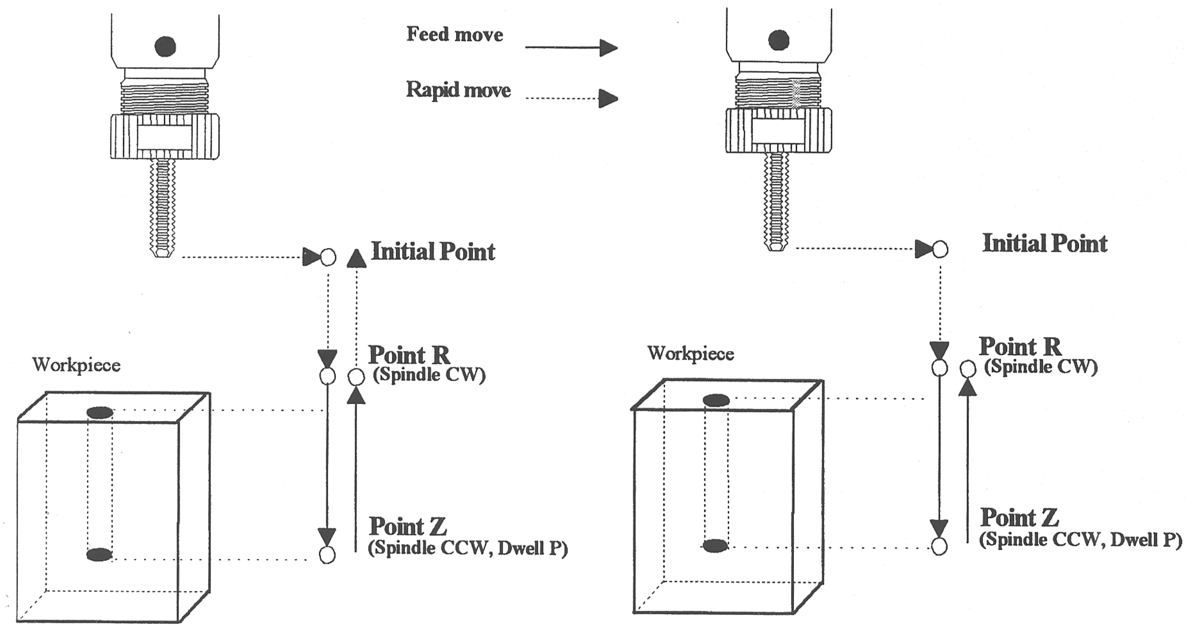
G83 Using G99

G83 is a deep hole drilling cycle. It periodically retracts the tool to the surface to clear accumulated chips, then returns to resume drilling where it left off. The retract and return are performed at the rapid rate. Because there may be chips in the bottom of the hole, the tool does not return all the way to the bottom at the rapid rate. Instead it slows to feedrate a short distance above the bottom. This clearance distance is selected by setting Parameter 83 with G10 (see example below).

Example:

```
G10 P83 R.05                ; set clearance to .05"
G83 X0 Y0 R.1 Z-2 Q.5       ; drill 2" deep hole in 0.5" steps
G80                         ; cancel canned cycle
```

G84 - Tapping



G84 using G98 **G84 using G99**

G84 performs right-hand tapping. The spindle speed (and feedrate, if you are doing floating tapping) should be set and the spindle started in the CW direction before issuing G84. G84 will normally use the default M4 to select spindle CCW (at the bottom of the hole) and M3 to re-select spindle CW (after backing out of the hole) depending on the settings of parameters 74 and 84.

The tap may continue to cut a short distance beyond the programmed Z height as the spindle comes to a stop before reversing. When tapping blind holes, be sure to specify a Z height slightly above the bottom of the hole to prevent the tool from reaching bottom before the spindle stops.

Note: If rigid tapping is enabled, a Q may be used to set the thread lead or pitch. However, because Q is not modal in the case of Rigid Tapping, you must specify Q on every line at which Rigid Tapping is to occur.

Note: At the bottom of the hole, G84 will call the default version of the specified M function even if it has been customized by an M function macro.

WARNING FEED HOLD is temporarily disabled during the tapping cycle, but it will be re-enabled at the end of the cycle.

NOTICE Pressing CYCLE CANCEL while the tap is in the hole will very probably break the tap or strip the threads in the tap hole. However, do so if it is an emergency.

Example:

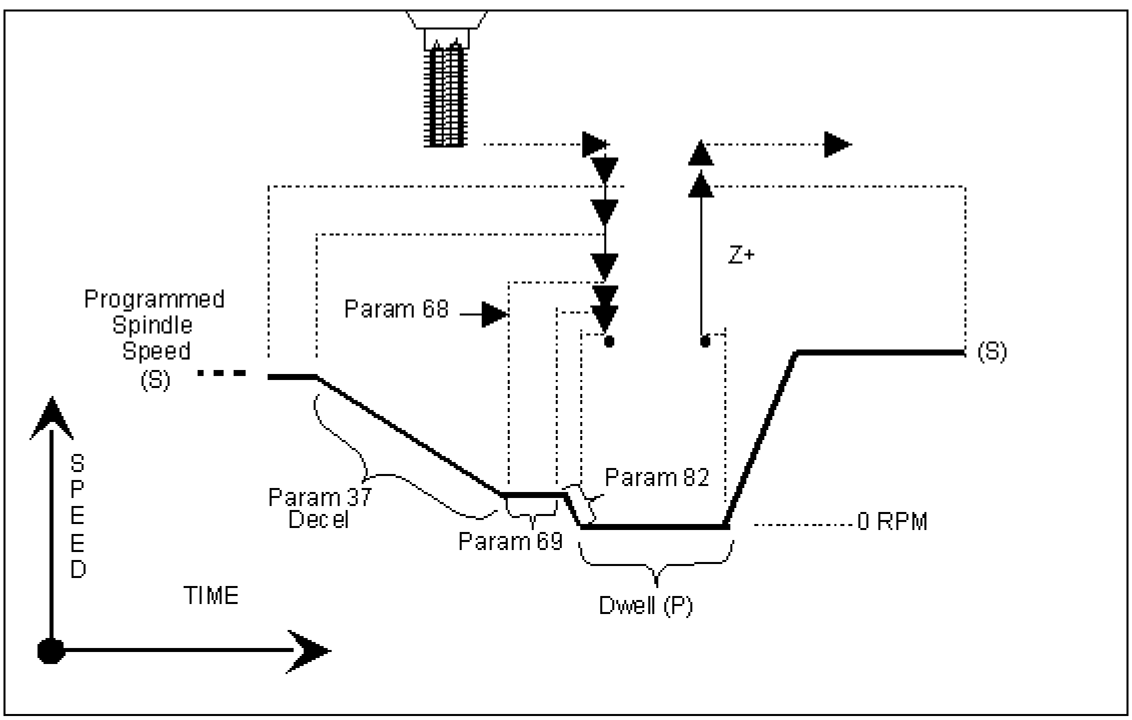
```
M3 S500 F27.78              ; start spindle CW, set up for 18 pitch tap
G84 X1 Y1 R.1 Z-.5          ; tap a 0.5 deep hole at X1 Y1
Y1.5                        ; ... and another one at X1 Y1.5
G80                         ; cancel canned cycle
…
; Using Tool 15 Rigid Tap 6-32
G84 X0.337 Y0.925 Q0.03125 Z-0.35 R0.1       ; tap first hole
G84 X3.312 Y0.925 Q0.03125 Z-0.35 R0.1       ; tap second hole, must use Q
G80                                          ; cancel canned cycle
…
; Using Tool 22 Rigid Tap 4-40
G84 X1.862 Y1.627 Q0.025 Z-0.19 R0.1         ; tap first hole
G84 X2.862 Y1.627 Q0.025 Z-0.19 R0.1         ; tap second hole, must use Q
G84 X4.262 Y1.627 Q0.025 Z-0.19 R0.1         ; tap third hole, must use Q
G80                                          ; cancel canned cycle
```

Tech Tip – How to Setup Rigid Tapping

Overview

This section describes the theory of rigid tapping parameters, to control accuracy of depth of cut and quality of threads, in various working materials.

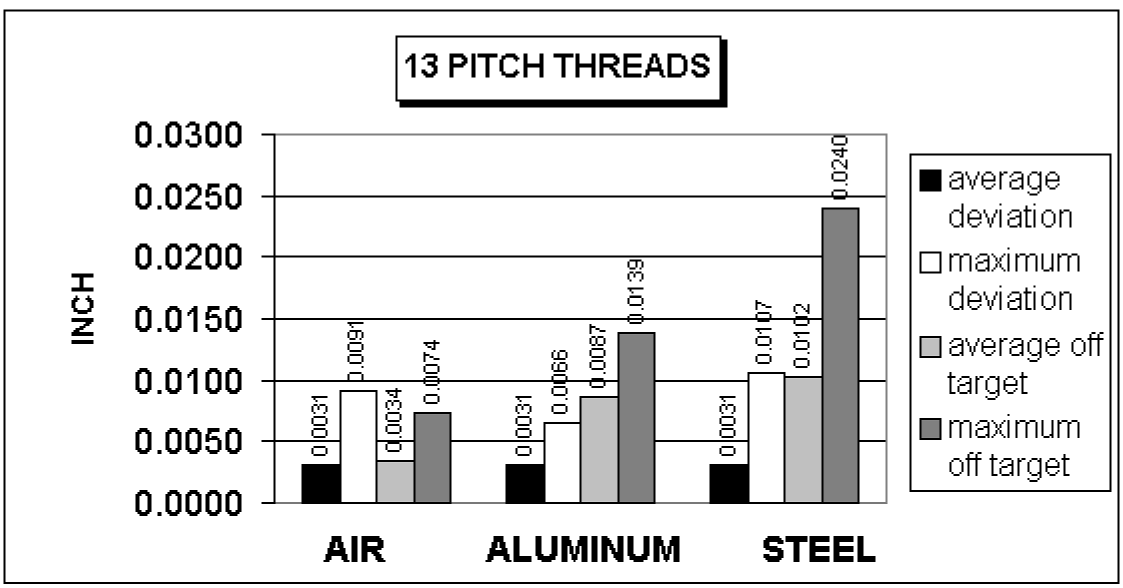
Graphic representation of parameter controls



List of Rigid tapping setup parameters – see Chapter 14 for more details

Parameter	Function
34	Spindle Encoder Counts/Rev
35	Spindle Encoder Axis Number
36	Rigid Tapping Enable/Disable
37	Spindle Deceleration Time
68	Minimum Rigid Tapping Spindle Speed
69	Duration For Minimum Spindle Speed
74	M-Function executed at bottom of tapping cycle
84	M-Function executed at return to initial point of tapping cycle
82	Spindle Drift Adjustment

Graphic representation of test results for precision



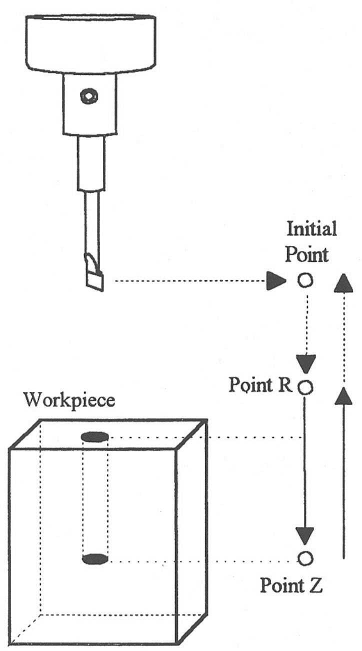
The above charts show test results of rigid tapping, utilizing version 7.14 software. The tool used in the testing was a ½-13 spiral fluted tap with TiN coating. Coolant used was water base soluble oil. Hole size was .4218. Tapping depth was .800. Also note that the parameters were adjusted to cut air, and not changed for aluminum or cold rolled steel for these tests. It can be seen, as the material changes, so does the off target values. This is due in part to the amount of torque required from the spindle to cut the various types of material. For testing purposes, the parameter settings for the above results were as follows.

Parameter 36 = 1, Parameter 37 = 3, Parameter 68 = 100, Parameter 69 = 1.25, Parameter 82 = 108

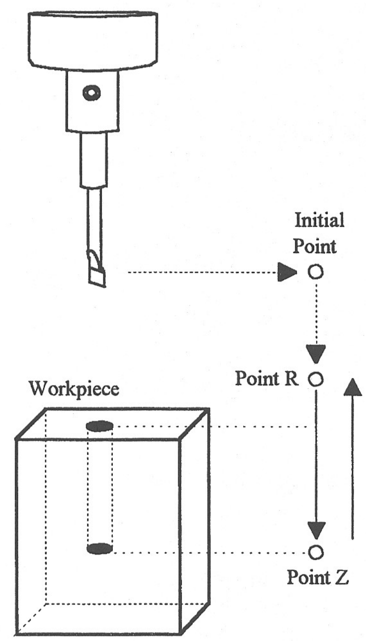
Summary

Rigid tapping parameters will vary from machine to machine. Not all machines are built the same (i.e. Spindle hp, inverter type, rigidity, etc.), and tooling will play a roll in performance also. It was found through our testing, if we changed one physical parameter, (i.e. using a tapping oil instead of water base coolant), it improved the off target values by 1.5%. This is due to the fact that less friction is present when using special cutting oil, therefore requiring less hp by the spindle to drive the tap. In most cases, rigid tapping depths should be able to be held within +/- .008 inch or less by adjusting parameter 82 for specific cases.

G85 – Boring



G85 Using G98



G85 Using G99

G85 is similar to G81, except that the tool is retracted with a feedrate move instead of a rapid move. G85 may be used for tapping with reversing tap heads such as the Tapmatic NCR series.

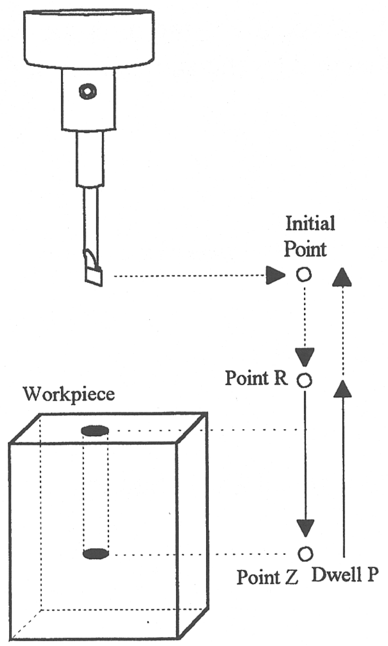
Example 1:

```
G85 X1 Y1 R.1 Z-.5            ; bore a 0.5" hole at X1 Y1
G80                           ; cancel canned cycle
```

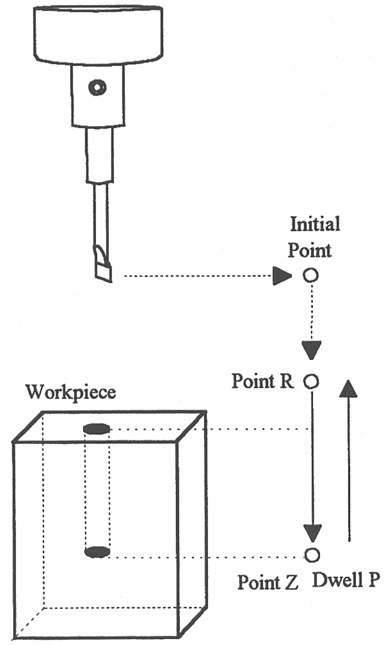
Example 2:

```
M3 S500 F27.78                ; start spindle CW, set for 18 pitch tap
M109/1/2                      ; disable feedrate and spindle overrides
G85 X1 Y1 R.1 Z-.4            ; tap hole at X1 Y1 to a depth of 0.4"
M108/1/2                      ; enable feedrate and spindle overrides
G80                           ; cancel canned cycle
```

G89 - Boring cycle with dwell



G89 Using G98



G89 Using G99

G89 is similar to G85, except that it includes an optional dwell at the bottom of the hole before retracting the tool.

Example:

```
G89 X1 Y1 R.1 Z-.5 P.1       ; bore 0.5" hole at X1 Y1, dwell .1 seconds
G80                          ; cancel canned cycle
```

G90 & G91 - Absolute/Incremental Positioning Mode

G90 selects absolute positioning, and G91 selects incremental positioning. In absolute positioning, all coordinates are relative to the origin (0,0,0,0). In incremental positioning, all coordinates are distances relative to the last point.

G90 Absolute positioning
G91 Incremental coordinates

Example:

```
G90 X2 Y3               ; moves the X and Y axes from the current position
                        ; to X2, Y3.
G91 X1 Y0               ; moves the X axis 1 inch referenced from the last X
                        ; position, the Y axis does not move.
```

G92 - Set Absolute Position

G92 sets the current absolute position to the coordinates specified. This command only affects the currently set Work Coordinate System.

Example:

```
G0 X5 Y3 Z-2 W5         ; Moves to the specified location
G92 X1 Y0 Z0 W1         ; Sets the current position to the absolute
                        ; position specified.
```

G93 - Inverse Time

Rather than using a conventional federate in Inch per Minute or MM per Minute, F in inverse time mode specifies the movement frequency for subsequent moves. Specifically, the inverse time feedrate is the inverse of the amount of time that a move is allowed to take.

For example, using inverse time, the block:

```
   G1 G93 X10 F1
```

takes 1 minute to cut, regardless of X length. At F2.0 (1/2.0 minute) it takes 30 seconds; At F3.0 it takes 20 seconds, and so on.

G93 is Modal, and remains in effect until a G94 is issued to cancel the G93.

Example of use in a program:

```
G0 G54 G90 X2.2126 Y-1.1995 A94.75 B-.161 S3000 M3  ; Move to start
G43 H26 Z7.0002 M8
Z3.1002
G1 G93 X2.2048 Y-1.2593 Z3.0204 F100.               ; Enable Inverse Time
X2.2052 Y-1.2586 Z3.0202 A94.756 B-.162 F30000.     ; 1/30000 min for move
X2.2079 Y-1.2578 Z3.0197 A94.763 B-.173 F30000.     ; 1/30000 min for move
X2.2124 Y-1.2566 Z3.0189 A94.773 B-.189 F30000.     ; 1/30000 min for move
X2.2184 Y-1.2551 Z3.0179 A94.786 B-.212 F24065.56   ; 1/24065 min for move
X2.2258 Y-1.2533 Z3.0167 A94.802 B-.24 F19736.14    ; 1/19736 min for move
X2.2345 Y-1.2512 Z3.0155 A94.819 B-.272 F17187.45   ; 1/17187 min for move
```

G93.1 – Velocity Scrubber for Smoothed Inverse Time Data

This special feedrate interpretation mode substitutes inverse time feedrates (usually posted from a CAD/CAM program) with optimized feedrates to ensure that the tool tip center is moved at a set feedrate in physical 3D space, taking into account tool length offset and the machine geometry as set by parameters 116-119 (see Chapter 14) . The set feedrate (F) can be on the same G-code line as the G93.1 or can be the last modal feedrate specified on prior G-code lines.

This mode is intended for CNC jobs run on machines configured with a triangular rotary 5th axis such as a Tilt Table or Articulated Head (see parameter 166 in Chapter 14), although that is not an enforced definite requirement to turn on this mode. It is recommended that moves programmed with this mode should be in small vectors, and any long moves that require a sweep of a rotary axis by more than 10 degrees should be broken up into 2 or more smaller moves. Note that Smoothing must also be turned on (P220 = 1) for this feature to work; otherwise, it will be treated as a regular G93 (Inverse Time).

Example of use in a program (based on the example from G93 above):

```
G0 G54 G90 X2.2126 Y-1.1995 A94.75 B-.161 S3000 M3  ; Move to start
G43 H26 Z7.0002 M8
Z3.1002
G1 G93.1 X2.2048 Y-1.2593 Z3.0204 F150. ; Force tool tip to move 150 in/min
X2.2052 Y-1.2586 Z3.0202 A94.756 B-.162 F30000.   ;Ignore F..Move at 150ipm
X2.2079 Y-1.2578 Z3.0197 A94.763 B-.173 F30000.   ;Ignore F..Move at 150ipm
X2.2124 Y-1.2566 Z3.0189 A94.773 B-.189 F30000.   ;Ignore F..Move at 150ipm
X2.2184 Y-1.2551 Z3.0179 A94.786 B-.212 F24065.56 ;Ignore F..Move at 150ipm
X2.2258 Y-1.2533 Z3.0167 A94.802 B-.24 F19736.14  ;Ignore F..Move at 150ipm
X2.2345 Y-1.2512 Z3.0155 A94.819 B-.272 F17187.45 ;Ignore F..Move at 150ipm
```

G94 - Cancel Inverse Time

G94 is used to cancel Inverse Time feedrates, and return to regular Feed per Minute feedrates.

G98 - Initial Point Return

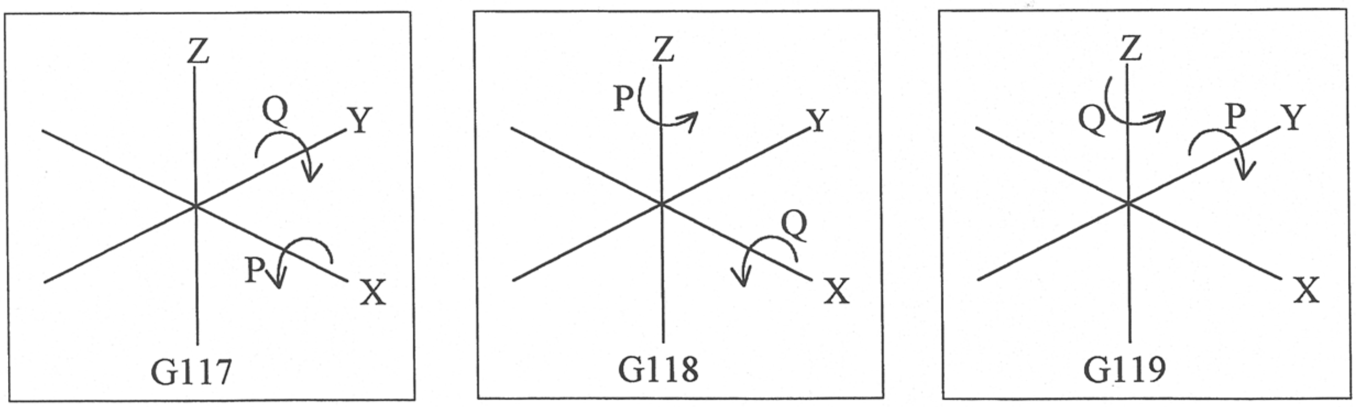
G98 sets the +Z return level to point I as pictured in Figure 1 in the Canned Cycle Section.
(G98 is the default setting)

G99 - R Point Return

G99 sets the +Z return level to point R as pictured in Figure 1 in the Canned Cycle Section.

G117, G118, G119 - Rotation of Pre-set Arc Planes

G117, G118 and G119 have the same functionality as G17, G18 and G19, respectively, except that they include 2 optional parameters P and Q to specify the arc plane rotation away from the pre-set arc plane: P specifies the arc plane angle of rotation (in degrees) around the first axis and Q specifies the arc plane angle of rotation around the second axis.



For the G117 plane, the "first axis" is X and the "second axis" is Y.
For the G118 plane, the "first axis" is Z and the "second axis" is X.
For the G119 plane, the "first axis" is Y and the "second axis" is Z. If P and/or Q are not specified, the angles are assumed to be 0 degrees. If both P and Q parameters are 0, then the plane is assumed to be an orthogonal (pre-set) arc plane. The center of the arc can be specified by the user in a 3D form both in G17-G19 and in G117-G119 (all I, J, K values are allowed at the same time with G2 and G3). Any arc center component outside the circular plane is ignored.

Example:

```
G00 X0 Y0 Z1                                ; rapid move
G03 G18 X1 Y0 Z0 K-1 F20                    ; arc mill
G00 X0 Y0 Z1.1                              ; retract move
G01 Z1                                      ; move to start of contour
G03 G118 P1.000000 X0.9998 Y0.0175 Z0 K-1   ; arc mill rotated about Z
```

* NOTE: G117-G119 will not be permitted while cutter compensation is turned on. Also, scaling is not allowed while G117-G119 is specified and G117-G119 is not allowed while scaling is active.

* NOTE: G117-G119 will not work when Smoothing is turned on (P220 = 1).

G173, G174, G176, G181, G182, G183, G184, G185, G189 – Compound Canned Cycles

On a machine configured as an Articulated Head machine with the TWCS feature enabled, the Compound Canned Cycle G-codes are used to perform tilted-head Drill/Bore/Tap operations when the currently selected WCS is not transformed (TWCS=No). See Parameter 166 in Chapter 14 for information on Articulated Head configuration and turning on the TWCS feature for the machine.

On a such a machine, Compound Canned Cycles are used as an alternative to the their corresponding non-compound Canned Cycles because the non-compound versions cannot be used when the currently selected WCS is not transformed (TWCS=No) and the spindle head is tilted off vertical (B/5th axis angle is not 0). In all other situations, Compound Canned Cycles are the same in functionality as their corresponding non-compound counterparts.

Compound Canned Cycle G-code	Function	Corresponding non-compound Canned Cycle G-code
G173	Compound High Speed Peck Drilling	G73
G174	Compound Counter Tapping	G74
G176	Compound Fine Bore Cycle	G76
G181	Compound Drilling and Spot Drilling	G81
G182	Compound Drill with Dwell	G82
G183	Compound Deep Hole Drilling	G83
G184	Compound Tapping	G84
G185	Compound Boring	G85
G189	Compound Boring with Dwell	G89

G180 – Cancel Canned Cycles

G180 has the exact same functionality as G80.

Chapter 13
CNC Program Codes: M functions

M functions are used to perform specialized actions in CNC programs. Most of the M-series Control M functions have default actions, but can be customized with the use of macro files.

Summary of M functions

M00 Stop for Operator	M80 (macro) Carousel In *
M01 Optional Stop for Operator	M81 (macro) Carousel Out *
M02 Restart Program	M91 Move to Minus Home
M03 Spindle On Clockwise	M92 Move to Plus Home
M04 Spindle On Counterclockwise	M93 Release/Restore Motor Power
M05 Spindle Stop	M94,M95 Output On/Off
M06 Tool Change	M98 Call Subprogram
M07 Mist Coolant On	M99 Return from Macro or Subprogram
M08 Flood Coolant On	M100 Wait for PLC bit (Open, Off, Reset)
M09 Coolant Off	M101 Wait for PLC bit (Closed, On, Set)
M10 Clamp On	M102 Restart Program
M11 Clamp Off	M103 Programmed Action Timer
M14 (macro) Swing Arm Pot Up *	M104 Cancel Programmed Action Timer
M15 (macro) Unclamp Tool with air on *	M105 Move Minus to Switch
M16 (macro) Unclamp Tool with air off *	M106 Move Plus to Switch
M17 (macro) Prepare for Tool Change *	M107 Output Tool Number
M18 (macro) Home Tool Changer *	M108 Enable Override Controls
M19 (macro) Orient Spindle *	M109 Disable Override Controls
M20 (macro) Pickup Tool *	M115,M116,M125,M126 Protected Move Probing Functions
M21 (macro) Move Head Up *	M120 Open data file (overwrite existing file)
M22 (macro) Move Head to ATC level *	M121 Open data file (append to existing file)
M23 (macro) Rotate Carousel *	M122 Record local position(s) in data file
M24 (macro) Start Tool Put-back *	M123 Record value and/or comment in data file
M25 Move to Z Home	M124 Record machine position(s) in data file
M26 Set Axis Home	M127 Record Date and Time in a data file
M30 Customizable M-code for End of Intercon Program	M128 Move Axis by Encoder Counts
M39 Air Drill	M129 Record Current Job file path to data file
M41 (macro) Select Low Spindle Gear Range *	M130 Run system command
M42 (macro) Select Medium-Low Spindle Gear Range *	M150 Zero Spindle Encoder at Next Index
M43 (macro) Select High Spindle Gear Range *	M200/M201 Stop for Operator, Prompt for Action
M50 (macro) Index Tool Plus *	M223 Write Formatted String to File
M51 (macro) Index Tool Minus *	M224 Prompt for Operator Input Using Formatted String
M60 (macro) 5 Axis Digitizing *	M225 Display Formatted String for A Period of Time
	M290 Digitize Profile
	M300 Fast Synchronous I/O update
	M333 Axis Role Re-assignment
	M1000-M1015 Graphing Color for Feedrate movement

* M functions marked with "(macro)" actually have no standard default action, and could possibly be unimplemented and therefore unavailable on your machine. Also, their stated function is only standard on certain machines.

Macro M functions (custom M functions)

Most M-Series CNC M functions from 0 through 90 can be fully customized. Exceptions are M2, M6, and M25 that can be customized, but will always move the 3rd (Z) axis to the home position before executing the macro M function commands. No M functions above 90 may be customized with macros. The default action listed will be performed unless that M function has been customized.

To create a macro for an M-function, a file must be created in the C:\cncm directory. The file's name must be mfuncXX.mac where XX is the M function number used to call the macro. M functions 0-9 must use single digits in the filename (e.g. Use mfunc3.mac, **not** mfunc03.mac). The contents of the file may be any valid M and G codes.

Each time the M function is encountered in a program, the macro file will be processed line by line.

* NOTE: Nesting of macro M functions is allowed. Recursive calls are not: if a macro M function calls itself, the default action of the function will be executed.

Example: Turn on spindle with variable frequency drive and wait for "at speed" response. This example depends on M94/M95 1 being used for the Spindle Enable signal and input 5 being the spindle at speed signal in the PLC program.

Create file c:\cncm\mfunc3.mac with contents as follows:

```
M94/1          ; request spindle start
M101/50005     ; wait for up to speed signal
```

M2, M6, and M25 **always** move the Z-axis to the home position before any other motion. All other M functions are performed after the motion of the current line is complete.

The M and G codes in a macro M function are not usually displayed on the screen as they are executed, and are all treated as one operation in block mode. If you wish to see or step through macro M functions (e.g. for testing purposes), set parameter #10 as follows:

0 Don't display or step through macro M functions
1 Display macro M functions, but don't step through them
2 Display and step through macro M functions

* NOTE: You cannot use block mode to step through a macro M function called using the G81 transformation (see Chapter 12); the action timer will expire before you can press **CYCLE START**.
NOTE: Only one M function per line is permitted.

* NOTE: Unlike subprograms invoked with G65, macro arguments passed into a macro M function are passed in by reference. This means local changes to #1 through #33 or #A through #Z will change the their values upon return from the macro M function.

M00 - Stop for Operator

Motion stops, and the operator is prompted to press the **CYCLE START** button to continue.

M01 - Optional Stop for Operator

M1 is an optional pause, whose action can be selected by the operator.

When optional stops are turned on, M1 will pause the currently running job until **CYCLE START** is pressed. However, if optional stops are turned off, M1 will not pause the program.

*NOTE: If you plan to override the default action of M1 with a macro file, you may want to include a call to M1 within the macro file so that the default actions of M1 will still be effective in the overridden M1. Otherwise, if a call to M1 is not included within the macro file, the new overridden M1 will cause optional stops to be ineffective.

M02 - Restart Program

M2 moves the Z-axis to the home position, performs any movement requested, and restarts the program from the first line. The operator is prompted to press the **CYCLE START** button to continue.

M03 - Spindle On Clockwise

M3 requests the PLC to start the spindle in the clockwise direction.

Default action:

```
M95/2
M94/1
```

M04 - Spindle On Counterclockwise

M4 requests the PLC to start the spindle in the counterclockwise direction.

Default action:

```
M95/1
M94/2
```

M05 - Spindle Stop

M5 requests the PLC to stop the spindle.

Default action if the spindle had been spinning CW:

```
M95/2
M95/1
```

Default action if the spindle was OFF or was spinning CCW:

```
M95/1
M95/2
```

M06 - Tool Change

M6 moves the Z-axis to the home position and stops the spindle and coolant. If an automatic tool changer is installed, it then commands the tool changer to switch to the requested tool. Otherwise, it prompts the operator to insert the tool and then press the **CYCLE START** button on the Operator Panel.

Default action (no tool changer):

```
M25             ; always does M25 first
M95/1/2/3/5     ; turn off spindle & coolant
M100/75         ; wait for CYCLE START button
```

Default action (tool changer installed):

M25	; always does M25 first
M95/1/2/3/5	; turn off spindle & coolant
M95/16	; turn off tool changer strobe
M107	; send tool number to tool changer
M94/16	; turn on tool changer strobe
M101/32	; wait for acknowledge from changer
M95/16	; turn off tool changer strobe
M100/32	; wait for acknowledge from changer

Manual tool changes are selected by setting Parameter 6 to 0 in the Machine Parameters table. The automatic tool changer is selected by setting Parameter 6 to 1 (see Chapter 14).

The PLC program must be involved in commanding an automatic tool changer and its associated strobe, BCD and Acknowledge lines. See Chapter 5 of the service manual for details of how such a PLC program could be constructed.

M07 - Mist Coolant On

M7 causes the PLC to start the mist coolant system.

Default action:
M95/3
M94/5

M08 - Flood Coolant On

M8 causes the PLC to start the flood coolant system.

Default action:
M95/5
M94/3

M09 - Coolant Off

M9 causes the PLC to stop the coolant system.

Default action:
M95/3/5

M10 - Clamp On

M10 causes the PLC to activate the clamp.

Default action:
M94/4

Note: adding 256 to parameter 178 can switch M10 and M11.

M11 - Clamp Off

M11 causes the PLC to release the clamp.

Default action:
M95/4

M17 – Prepare for Tool Change (Macro)

M17 has no default action, therefore a custom M17 macro must be defined for this feature to work. If defined, the M17 macro turns off spindle and coolant and starts the spindle orientation process in preparation for M6 (Tool Change). The M17 macro is allocated for use with Intercon and the setting in Parameter 162. See Parameter 162 in Chapter 14 for more information.

M19 – Spindle Orient (Macro)

M19 has no default action, therefore a custom M19 macro must be defined for this feature to work. If defined, the M19 macro sends a request to the PLC to rotate the spindle to its pre-set orient position.

M25 - Move to Z Home

By default, M25 moves the Z axis to the home position at the Z axis maximum rate. The Z axis home position is the Z axis component of the Return #1 (G28) machine position. (The Return #1 position is first machine coordinate position defined in the Return sub-menu of the Work Coordinate System Configuration.)

The default action of M25 only involves the Z axis. However, if you specify axis arguments (up to 3), then those axes specified will be moved to their axis home positions (Return #1 machine position).

Examples:

```
M25          ; move Z to home
M25 /Z       ; same as M25 by itself –  move Z to home
M25 /X/Y/Z   ; move X, Y and Z to their Return #1 positions
M25 /X       ; move only X to its Return #1 position
```

M26 - Set Axis Home

M26 sets the machine home position for the specified axis to the current position (after the line's movement). If no axis is specified, M26 sets the Z-axis home position.

Example:

```
M91/X        ; home X axis to minus home switch
M26/X        ; set machine home for X-axis there
M92/Z        ; home Z-axis to plus home switch
M26          ; set machine home for Z-axis there
```

M30 - Custom M Code

Intercon posts an M30 at the end of every G code program. By default M30 performs no operation. If you wish to perform certain operations at the end of every program, this M code can be customized to meet you needs. For more information on customizing M codes see the beginning of this chapter.

M39 - Air Drill

M39 is a default air drill activation sequence with a timeout. The sequence of operations is as follows:

```
M94/15       ; activate M function request 15
M103/2       ; start 2 second timer
M100/50015   ; wait for input 15 to open
M95/15       ; deactivate M function request 15
M104         ; cancel timer
```

* NOTE: This program will be canceled by timer expiration if input 15 does not open within 2 seconds after M function request 15 is activated. The PLC program must be involved in taking away the drill output when the CNC program stops:

Example:

```
; PLC program fragment
CNC_program_running is SV_PROGRAM_RUNNING   ;program running indicator
M15 is SV_M94_M95_15                        ; M function 15 indicator
drill_out is OUT5                           ; air drill output relay
if M15 &&  CNC_program running then (drill_out)  ; Drill On if M94/15 and the
                                                 ; CNC program is running. Drill
                                                 ; Off if M95/15 or the CNC
                                                 ; program is terminated.
```

M41, M42, M43 – Select Spindle Gear Range (Macros)

M41, M42, and M43 have no default actions, and therefore custom macros must be defined for these M codes in order to make this feature work. If defined, these macros notify the PLC of which spindle gear range is selected according to the following table:

Macro M function	Action
M41	Select Low Gear Range
M42	Select Medium-Low Gear Range
M43	Select High Gear Range

Note that selecting a "Medium-High" Gear Range is currently not supported by this schema, although that would not prevent a system intergrator from defining another custom macro M function to do that.

M60 – 5-Axis Digitizing Macro

The M60 is used only when digitizing with the 5-Axis Tilt Table system.

The M60 uses user variables to assign a "Start Position" and a "Finish Position" for 5-Axis Digitizing. When the M60 is executed, the probe will be moved from the start position to the end position. If during the motion the probe detects a surface, the surface position is recorded, and the probe returns to the start position. If no surface is found the probe returns to the start position.

The following variables are to be assigned before the M60 is called, the M60 will then use the positions as assigned by the user variables:

#29100 = X-Axis Probing vector start point (Initial Position)
#29101 = Y-Axis Probing vector start point (Initial Position)
#29102 = Z-Axis Probing vector start point (Initial Position)
#29103 = A-Axis Probing vector start point (Initial Position)
#29104 = B-Axis Probing vector start point (Initial Position)
#29110 = X-Axis Probing vector end point
#29111 = Y-Axis Probing vector end point
#29112 = Z-Axis Probing vector end point

Upon a successful probe the M60 will use an M122 to save the probed position to a text file that should be opened using an M120 or M121 before calling the M60

Example M60 use in a program:

```
M121 "m60test.dig5"     ; Open text file to record data too

#29100 = -8.7999 ; X-Axis Start Position

#29101 = .3747    ; Y-Axis Start Position

#29102 = -1.1832 ; Z-Axis Start Position

#29103 = 85.957   ; A-Axis Start Position

#29104 = 21.36    ; B-Axis Start Position

#29110 = -8.7138 ; X-Axis End Position

#29111 = -.0159   ; Y-Axis End Position

#29112 = -1.183   ; Z-Axis End Position

M60               ; Execute M60
```

M91 - Move to Minus Home

M91 moves to the minus home switch of the axis specified at the slow jog rate for that axis. After the minus home switch is tripped, the axis reverses until the home switch clears, and stops when it detects the index pulse.

Example:

```
M91/X        ; moves the X-axis to the minus home switch.
G92 X-10     ; sets X minus home switch at -10
```

M92 - Move to Plus Home

M92 moves to the plus home switch of the axis specified at the slow jog rate for that axis. After the plus home switch is tripped, the axis reverses until the home switch clears, and stops when it detects the index pulse.

Example:

```
M92/X        ; moves the X-axis to the plus home switch.
G92 X+10     ; Sets X plus home switch at +10
```

M93 – Release/Restore Motor Power

M93 releases or restores motor power for the axis specified. If no axis is specified, then all axes are released.

Example:

To release motor power:

```
M93/X        ; releases the X axis.
M93          ; releases the motors on all axes.
```

Example:

To restore power:

```
M93/X P1     ; restore power to the X axis motor.
M93 P1       ; restore power to the motors on all axes.
```

M94/M95 - Output On/Off

There are 128 user definable system variable bits that can be used to communicate with the PLC. M94 and M95 are used to request those system variable bits to turn on or off respectively. Requests 1-128 are mapped to the PLC as system variables SV_M94_M95_1 through SV_M94_M95_128 as shown in the following table:

On	Off	PLC bit
M94/1	M95/1	SV_M94_M95_1
M94/2	M95/2	SV_M94_M95_2
M94/3	M95/3	SV_M94_M95_3
M94/4	M95/4	SV_M94_M95_4
. . .	. . .	. . .
M94/128	M95/128	SV_M94_M95_128

To use M94 and M95 to control a function external to the servo control, such as an indexer, the input request must be mapped to one of the PLC outputs in the PLC program. See M94/M95 function usage in the PLC section of the service manual.

Example:

M94/5/6 ; turns on SV_M94_M95_5 and SV_M94_M95_6.

* NOTE: M94 and M95 will cause prior motion to decelerate to a stop before the requested bits are turned on or off.
* NOTE: Requests 1-5, 15, and 16 are controlled by the default actions of M3, M4, M5, M6, M7, M8, M9, M10, M11, and M39. To override or disable a bit used in one of these M codes, define a custom M-function.

M98 - Call Subprogram

M98 calls a user-specified subprogram. A subprogram is a separate program that can be used to perform a certain operation (e.g. a drilling pattern, contour, etc.) many times throughout a main program.

Calling methods:

M98 P*xxxx* L*rrrr*

Or

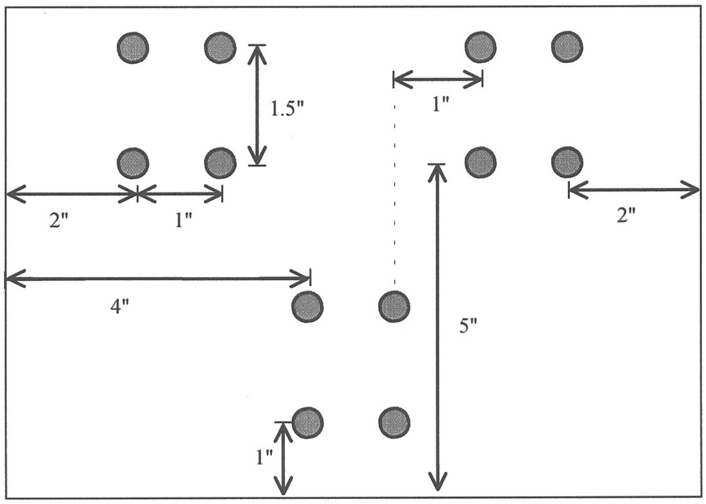
M98 "program.cnc" L*rrrr*

where *xxxx* is the subprogram number (referring to file O*xxxx*.cnc, 9100-9999 allowed, leading 0's required in filename, capital O, lowercase .cnc), *rrrr* is the repeat value, and "program.cnc" is the name of the subprogram file.

Subprograms are written just like normal programs, with one exception: an M99 should be at the end of the subprogram. M99 transfers control back to the calling program.

Subprograms can call other subprograms (up to 20 nested levels of calling may be used), Macro M-functions, and Macros. Macro M-functions and Macros can similarly call subprograms.

Subprograms 9100-9999 can also be embedded into a main program, using O9xxx to designate the beginning of the subprogram and M99 to end it. The CNC software will read the subprogram and generate a file O9*xxx*.cnc. The CNC will not execute the subprogram until it encounters M98 P9*xxx*.
NOTE: An embedded subprogram definition must be placed before any calls to the subprogram.



(0,0)

Suppose that a drilling pattern of 4 holes is needed in 3 different locations.

This subprogram would handle the drilling and incremental moves between the holes:

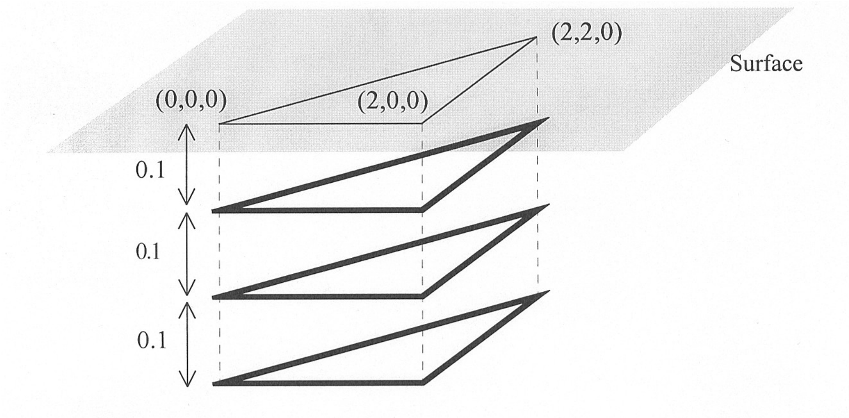
```
O9101                       ;Program O9101.cnc
G91 F10                     ;Incremental positioning
G81 X0 Y0 R -.4 Z-.6        ;Drill lower left hole
Y1.5 R -.4 Z-.6             ;Drill upper left hole
X1 R -.4 Z-.6               ;Drill upper right hole
Y-1.5 R-.4 Z-.6             ;Drill lower right hole
G80                         ;Cancel canned cycles
M99                         ;End of subprogram
```

The main program would call this subprogram three times:

```
:Main program
G90 G0 X2 Y5 Z0.5           ;Move to first hole pattern
M98 P9101 L1                ;Call subprogram O9101.cnc
G90 G0 X4Y1 Z0.5            ;Move to second hole pattern
M98 P9101 L1                ;Call subprogram
G90 G0 X6 Y5 Z0.5           ;Move to third hole pattern
M98 P9101 L1                ;Call subprogram
:End program
```

Another example is "looping" or consecutively repeating a section of code. Here, the subprogram will be part of the main program.

```
:Main program
G90 G0 X0 Y0 Z0.1
G1 Z0 F30
O9100                       ;Beginning of subprogram
G91 G1 Z-0.1 F5
G90 X2 F30
Y2
X0 Y0
M99                         ;End of subprogram 9100
M98 P9100 L3                ;Repeat O9100 3 times
M25 G49                             ;End main program
```



M99 - Return from Macro or Subprogram

M99 designates the end of a subprogram or macro and transfers control back to the calling program when executed. M99 may be specified on a line with other G codes. M99 will be the last action executed on a line. If M99 is not specified in a subprogram file, M99 is assumed at the end of the file:

Example:

```
G1 X3 M99                    ;Move to X3 then return to calling program.
```

If M99 is encountered in the main job file, it will be interpreted as the end of the job. If M99 is encountered in an M function macro file, it will be interpreted as the end of any enclosing subprogram or macro, or as the end of the job.

M100 - Wait for PLC bit (Open, Off, Reset)

M101 - Wait for PLC bit (Closed, On, Set)

The M100/M101 commands wait for a PLC bit to reach a state as indicated in the table below.

Number	PLC bit	M100	M101
50001 – 51312	INP1 – INP1312	open	closed
60001 – 61312	OUT1 – OUT1312	off	on
70001 – 71024	MEM1 – MEM1024	reset	set
90001 – 90064	T1 – T64 status bits	reset (not expired)	set (expired)
93001 – 93256	STG1 – STG256 status bits	reset (disabled)	set (enabled)
94001 – 94256	FSTG – FTSG256 status bits	reset (disabled)	set (enabled)
The number ranges 1-240 can be used to reference the first eighty INP, OUT, or MEM bits. It is recommended that existing CNC10 programs and macros be converted to the new ranges for use with CNC11.			
1 – 80	INP1 – INP80	open	closed
81 – 160	OUT1 – OUT80	off	on
161 – 240	MEM1 – MEM80	reset	set

Example:

```
M101/50001  ; wait for INP1 to close
M100/60002 ; wait for OUT2 to turn off
M101/70123 ; wait for MEM123 to be set (1)
```

NOTE: The numbers assigned to the PLC bits (except 1-240) are the same as those that can be used when referencing system variables in M- and G-code programs.

M102 - Restart Program

M102 performs any movement requested, and restarts the program from the first line. The Z-axis is **NOT** moved to the home position, and the operator is **NOT** prompted to press the **CYCLE START** button to continue.

M103 - Programmed Action Timer

M103 is used to set up the time limit for a timed operation. If the timer is canceled (usually by M104) before the specified time limit , the program will be canceled and the message "Programmed action timer expired" will be displayed. If another M103 is issued before the time limit expires, then this time limit is nullified and the new time limit will be set up as specified by the latest occurring M103. Note also that if M0 or M1 causes the program to stop momentarily and the “M0 jogging” feature is enabled, then the the timer will also be canceled without the need to issue M104.

Example: Activate a device and wait for a response. If there is no response within 4.5 seconds, cancel the program.

```
M94/12          ; turn on input request 12
M103/4.5        ; start 4.5 second timer
M100/4          ; wait for input 4 to open
M104            ; input 4 opened, cancel timer
```

M104 - Cancel Programmed Action Timer

M104 stops the timer started by the last M103 executed.

M105 - Move Minus to Switch

M105 moves the requested axis in the minus direction at the current feedrate until the specified switch opens (if the given P parameter is positive), or until the scecified switch closes (if P parameter is negative).

Example:

```
M105/X P5 F30   ; move the X axis in minus direction at 30"/min until
                ; the switch on INP5 opens
G92 X10         ; Sets X position to 10
M105/Z P-6      ; move the Z axis in minus direction until switch on INP6 closes
```

M106 - Move Plus to Switch

M106 moves the requested axis in the plus direction at the current feedrate until the specified switch opens (if the given P parameter is positive), or until the scecified switch closes (if P parameter is negative).

Example:

```
M106/Z P3 F30   ; move the Z axis in the plus direction at 30"/min, until
                ; the switch on INP3 opens
G92 X10         ; Sets Z position to 10
M106/X P-3      ; move the X axis in the plus direction until the switch on INP3 closes
```

M107 - Output Tool Number

M107 sends the current tool number to the automatic tool changer, via the PLC. M107 does not set the tool changer strobe or look for an acknowledgement from the changer (see M6).

Example:

```
M107            ; send request for tool to change
M94/16          ; turn on tool changer strobe
M101/5          ; wait for acknowledge on input 5
M95/16          ; turn off strobe
M100/5          ; wait for acknowledge to be removed
```

M108 - Enable Override Controls

M108 re-enables the feedrate override and/or spindle speed override controls if they were disabled with M109. A parameter of “1” indicates the feedrate override; “2” indicates the spindle speed override.

Example:

M109/1/2	; disables feedrate and spindle speed overrides
M108/1	; re-enables feedrate override
M108/2	; re-enables spindle speed override

M109 - Disable Override Controls

M109 disables the feedrate override and/or spindle speed override controls. It may be used before tapping with G85 to assure that the machine runs at the programmed feedrate and spindle speed. It is not necessary to specify M109 with G74 or G84; those cycles automatically disable and re-enable the override controls. M109 cannot be used in MDI mode.

Example:

M3 S500	; start spindle in clockwise direction, at 500 rpm
F27.78	; set feedrate for 18 pitch tap
M109/1/2	; disable feedrate and spindle speed overrides
G85 X0 Y0 R.1 Z-.5	; tap a hole
M108/1/2	; re-enable overrides

M115/M116/M125/M126* – Protected Move Probing Functions

The protected move probing functions provide the capability to program customized probing routines.

The structure for these commands is: **Mnnn** /*Axis pos* P*p* F*f L1*

Where:

nnn	is either 115, 116, 125, or 126.
Axis	is a valid axis label, i.e., X, Y, Z, etc.
pos	is an optional position
P	is a PLC bit number, which can be negative.
*F**	is a feedrate (in units per minute.)
*L1**	options for the M115/M116 commands that prevents an error if the probe does not detect a surface
Q1	is an option for M115/M116 that forces the DSP probe to move a “Recovery Distance” on retries. (See Machine Parameter 13 for “Recovery Distance”) ***Note:*** the Q1 option only applies for DSP Probes

For M115 and M116 functions, the indicated *axis* will move to *pos* (if specified) until the corresponding PLC bit *p* state is 1, unless *p* is negative, in which case movement is until the PLC bit state is 0(closed). A “*p* value” of 1 to 80 (or -1 to -80) specifies PLC bits INP1-INP80. Warnings are generated in the CNC software message window for "*Missing P value*" and "*Invalid P value*." If “*pos”* is not specified, M115 will move the *axis* in the negative direction, and M116 will move the *axis* in the positive direction. Note if “*pos”* is specified, then if does not matter whether M115 or M116 is used. Regardless of whether or not pos is specified, movement is bound by the settings in the software travel limits as well the maximum probing distance (Machine Parameter 16).

For M125 and M126 protected move functions, the behavior is identical to that of the M115 and M116 commands, except in regards to the PLC bit state. The M115 and M116 commands are to be used when one expects contact to be made and M125 and M126 commands are to be used when one does not expect any contact to be made.

Example:

Finding the center of a vertical slot. In this example, it is assumed that there is a probe connected to INP15 and that the probe tip is positioned somewhere in the slot, such that movement along the X-axis will cause a probe trigger.

```
M115/X P-15 F20        ; Move X minus at 20 ipm until probe trip
M116/X P15 F5          ; Move X plus at 5 ipm until probe clears
#100 = #5041           ; Record the point in user variable #100
M116/X P-15 F20        ; Move X plus at 20 ipm until probe trip
M115/X P15 F5          ; Move X minus at 5 ipm until probe clears
X[[#100+#5041]/2]      ; Move X to center of slot
```

*Usage is slightly different when using a DSP type probe. Please see below for dissimilarities between a standard DP4 probe and the DSP type probe.

*M115/M116/M125/M126 - DSP Probe specific information

Before attempting to use the protected move probing functions with a DSP type probe, please be sure to familiarize yourself with the DSP probe configuration in Chapter 9 of this manual. Using the protected probing moves with a DSP type probe may yield unexpected results if you do not fully understand the concepts and guidelines discussed in the DSP probe configuration section.

If the control is configured to use a DSP type probe, all M115/M116 moves will perform window checking and repeat on a failed window. On a failed window, a repeat attempt is made by returning to the starting point of the move.

Protected move probing functions follow the same command format as that of a standard probe (Mnnn /*Axis pos* P*p* F*f L1)* with the following exceptions:

f	*This will be ignored if "Force DSP Feedrate in M115/M116" has been set to yes.*
L1	*Still prevents a fault from occurring. Stores last DSP position on failed window.*
L2	*Like L1, prevents a fault from occurring but instead stores last mechanical pos. on failed window.*
Q1	*On a failed window, force a pull back distance equal to the Probing Recovery Distance (Parameter 13), instead of moving back to the starting point.*

DSP position vs. mechanical position:

Protected probing moves that are performed using a standard DP4 probe can collect only the point at which motion has stopped after detecting contact. This position is referred to as the "mechanical position". When using the DSP type probe, it detects and stores the contact position "on the fly". This position is in machine position (not a local WCS position) and is referred to as the DSP position.

Example from above – Modified to use a DSP type probe:

Finding the center of a vertical slot.

```
M115/X P-15                      ; Move X minus at DSP rate until probe trip (no feedrate needed)
#100 =[[#24301]-[#2500]]         ; Convert point to current WCS position, Store point in variable #100
M116/X P15                       ; Move X minus at 5 ipm until probe clears
M116/X P-15                      ; Move X plus at DSP rate until probe trip
X[[#100+[[#24301]-[#2500]]]/2]   ; Move X to center of slot
```

Retrieving the DSP position:

The last stored DSP position for axes 1-5 can be retrieved from system variables #24301-#24305 unless the L2 switch was used in which case #24301-#24305 will contain the mechanical position after a failed window.

M120 - Open data file (overwrite existing file)

This M function will open the requested data file for writing. If no drive or directory is specified with the file name, then the file will be opened in the same directory as the CNC program. If the file cannot be successfully opened, then an error will be returned, ultimately terminating the job. If a data file is already open when M120 is called, that file will first be closed, then the new file opened.

Example:

M120 "probetst.dat" ; Opens probetst.dat file to write data too

Note: M120 and M121 also allow use of the string user variables #300 - #399 to specify a filename. As an example, given that #300 = “myfile” and #301 = “cnc”

M120 “#300.#301” ;Opens the file “myfile.cnc” for data recording.

Keep in mind however that there is a quirk in the way that the M120/M121 operates that requires the '.' to be present so assigning #301 = “.cnc” and executing M120 “#300#301” does not work and generates a “Could not open file” error message.

M121 - Open data file (append to existing file)

This M function will open the requested file for writing at the end of the file. If no drive or directory is specified with the file name, then the file will be opened in the same directory as the CNC program. If the file does not already exist, it will be created. This is not an error. If the file cannot be successfully opened, then an error will be returned, ultimately terminating the job. If a data file is already open when M121 is called, that file will first be closed, then the new file opened.

Example:

M121 "c:\probetst.dat" ; Opens probetst.dat file to add data to it

String variables #300-#399 may also be used to specify a file name. Please see M120 above for details.

M122 - Record local position(s) and optional comment in data file

This M function will write the current expected position value to the data file, in the usual format (i.e. axis label before number, 4 decimal places in inch mode, 3 decimal places in millimeter mode. Any comment that appeared on the line with M122 will be outputted after the position(s). With no axis arguments, M122 will write the positions of all installed axes. With axis arguments, it will write the positions only of the requested axes. Positions will be written in local (not machine) coordinates, in native machine units. If no data file has been opened with M120 or M121 before M122 is called, then M122 will return an error and terminate the job. The parameter L1 may be used to suppress the new line character normally outputted after the last position. Furthermore, the output of axis labels, comma separators, and spaces can be enabled or suppressed via machine parameter 72 (see Parameter 72 in Chapter 14). If the control has been configured to use a DSP probe type, using parameter Q1 will write the values stored in #24301-#24305 to the file.

Examples (M function and sample output):

M122 ;comment	->	X1.2345 Y-3.2109 Z-0.5678 ;comment
M122 /X L1	->	X-1.5000
M122 /X	->	X-1.5000 X-2.0000

M123 - Record value and/or comment in data file

This M function will write the specified parameter value (if any) to the data file, followed by any comment that appeared on the line with M123. If a P value is specified, M123 will record the numeric value (4 decimal places in inches, 3 in millimeters). If neither a P value nor a comment was specified, M123 does nothing. This is not an error. If no data file has been opened with M120 or M121 before M123 is called, then M123 will return an error and terminate the job. The parameter L1 may be used to suppress the new line character normally outputted after the last value. The R and Q parameters can be used to specify the field width and precision, respectively. Furthermore, the output of axis labels, comma separators, and spaces can be enabled or suppressed via machine parameter 72 (see Parameter 72 in Chapter 14).

Examples (M function and sample output):

M123 ;1.2345	->1.2345
M123 P#A ; first macro argument	->1.2345 first macro argument
M123 Q0 P1.23	->1

M124 - Record machine position(s) and optional comment in data file

Identical to M122 above except that the m124 reports machine position instead of a local WCS position.

M127 - Record Date and Time in a data file

This M function is used to write the date, time, and year to the specified data file called out by the M120 or M121. Examples (M function and sample output): **Note: The M127 does not insert a semi-colon in front of the date.** If desired, use the M123 as shown below.

```
M121 "testdata.dat"
M123 ;;
M127
```

If you opened testdata.dat you would see: Day of week, Month, day, time, and year.
(i.e. ;Wed Aug 29 11:56:57 2007)

M128 – Move Axis by Encoder Counts

M128 moves the requested axis by L which specifies an encoder count position or quantity. The L parameter is subject to the current G90/G91 mode (absolute/incremental).

Example:

```
G91 M128/X L-5000          ; move the X axis incrementally by -5000 counts
```

M129 - Record Current Job file path to data file

This M function is used to write the current job's file path to the specified data file called out by the M120 or M121.

Example:

Run a job named job.cnc which contains the following 2 lines:

```
M121 "output.txt"
M129
```

If you opened the output.txt file you would see:

```
c:\cncm\ncfiles\job.cnc
```

M130 - Run system command

This allows shell commands to be called from a CNC program or MDI. M130 takes one string argument which contains the system command to execute.

For example:
M130 "mycommand.bat"
will run the batch file *mycommand.bat*.

Normally, the command will run asynchronously, meaning that the G-code program will not wait for the command to finish before continuing. However, if an L1 parameter is given, the command will prevent further G-code execution until a fault occurs (such as E-Stop).

M150 – Set Spindle Encoder to zero at next index pulse.

Formatted String Commands- M200, M223, M224, M225 & M290

The formatted string commands are provided to assist in custom screen and file I/O. A "*formatted-string*" is similar to the C programming language "printf" command, with various restrictions. The basic form of a *formatted-string* is a quoted string (comprised of a **single** line of up to 1024 characters) followed by a (possibly empty) list of user and/or system variable expressions. The variable expression is a '#' character followed by a number or bracketed expression.

For example, given #100 = 88* (ASCII 'X'), #300 = "absolute", and #101 = 1.2345, this string:

```
"The %c* axis %s position is %f" #100 #300 #101
```

evaluates to

```
"The X* axis absolute position is 1.23450"
```

The "%c"* is replaced by the ASCII character value of user variable #100, the "%s" is replaced by the string user variable #300, and the "%f" is replaced by the value of user variable #101.

Type specifiers
The 's', 'c', and 'f' are type specifiers, with 's' specifying a string user variable, 'f' specifying a floating point user variable, and 'c' specifying a single character substitution using the integer part of a floating point user variable. There should be one user variable expression for every '%' character in the quoted string. It is also possible to specify a field width by inserting a number between the '%' and the type specifier.

Example:
%20s – specifies that the substituted string is displayed in a field 20 characters long, right justified and padded with spaces on the left. Use "%-20s" for left justification.

The 'f' type can specify a precision such as:

- "%.4f" - display number rounded at the fourth decimal place.
- "%9.4f" - as above but in a field width nine characters wide.
- "%+9.4f" - as above with an '+' output if variable is positive.
- "%.0f" - display number rounded to integer

If no precision is specified, "%f" will use a default precision of the current DRO display precision.

Special characters
The quoted string may contain one or more "\n", each of which will be converted to a single newline character. Up to seven newlines can be specified in a single formatted string. However, a formatted string may not contain an embedded quote character '"' or other *printf*-style escape sequences such as '\t', '\\', or '\"'. If a quote character is desired, use a %c type specifier with a variable expression equal to 34.

User string variables #300-#399: These variables can be assigned a quoted string up to 80 characters in length and are retained until the CNC software is exited. For example,

```
#300 = “This is a text string of characters”
```

*The above method of representing an axis label should be used only when writing to an external file or for display in a message box. It is not valid if you are attempting to “build” a motion command in real-time from within the currently running g code program. If your intent is to use a variable to represent an axis label for a real-time command, you should instead use $ as the placeholder. The parser will replace a '$' character and the numerical expression following it with the ASCII character equivalent to the numerical expression, provided that it evaluates to the characters 'A' (65) through 'Z' (90). If the numerical expression is out-of-bounds, an “Invalid character” error occurs.

Example: Given #100 = 88, #101 = 1, #102 = 89, #103 = 2, and #104 = 10,
`G1 $[#100][#101] $[#102][#103] F[#104]` evaluates to `G1 X1 Y2 F10`

M200/M201 – Stop for Operator, Prompt for Action

M200 is used to pause the currently running job and prompt the operator for action. If M0_jogging is unlocked, or the control is in DEMO mode, jogging is enabled while waiting for the operator to respond. If this option has not been enabled, the behavior will default to that of a standard M0. (jogging disabled)

The syntax is: M200 *formatted-string [[user_var_expr] ...]*

Example:
M200 “Please jog the %c and %c axes to the desired X0, Y0 position\nPress Cycle Start to continue” #100 #101

M201 behaves exactly like M200 except that PLC bits SV_PROGRAM_RUNNING, SV_MDI_MODE, and SV_JOB_IN_PROGRESS are turned off while the prompt is displayed.

M223 – Write Formatted String to File

The M223 command writes a *formatted-string* to a file that was opened using the M120 or M121 commands. The syntax is:

M223 *formatted-string [[user_var_expr] ...]*

Example:
M223 “; The measured diameter of the pocket = %.4f\n” #100

M224 – Prompt for Operator Input Using Formatted String

The M224 command displays a *formatted-string and then accepts user input. The synta*x is:

M224 lvalue_*expr formatted-string* `[[user_var_expr] ...]`

Where *lvalue_expr* is a *user_var_expr that* evaluates to a user variable that can be written. If *lvalue_expr* is a string type (#300-#399) then the user input is assigned verbatim to the string. Otherwise, the user input is evaluated as any other “bracketed” numerical expression.

Example:

M224 #300 “Please enter the direction that you wish to probe in the %c axis: (+ or -)” #100

M225 – Display Formatted String for A Period of Time

The M225 command displays a *formatted-string* for a specified period of time. The syntax is:

M225 *time_expr formatted-string [user_var] ...*
where *time_expr* is a *user_var_expr* that evaluates to a floating point variable specifying the number of seconds to display the output, with a value of zero interpreted as indefinitely. The CYCLE_START key can be used to immediately continue running without waiting for the time to expire.

Example:
M225 #100 “Warning, %s is not selected\nPlease select %s and press Cycle Start to continue” #300 #300

M290 – Digitize Profile (Optional)

This performs a 2 axis digitize, probing along an axis while stepping over using a perpendicular travel axis. This M-code is similar to performing a single slice of Grid Digitizing with the Surface Following type selected (See Chapter 8). M290 expects that a file is already open with M120/121 (however, if not open, there will be no output).

The syntax is:
M290 /*a*__ #*vvv* /*b*__ #*vvvv* "*formatted-string*" Q__ R__ P__ L__

The first axis mentioned will be treated as the probing axis and the second will be treated as the travel axis.

<u>Explanation of M290 Arguments:</u>
/*a* +-nnn.nnn is the probing direction and max distance on axis “*a*”
/*b* +-nnn.nnn is the travel direction and max distance on axis “*b*” (perpendicular to probing axis)
#*vvv* and #*vvvv* (optional) are G-code variables that will be a receptacle of the <u>very last</u> probed position of the cycle.
"formatted-string" (optional) is the format of the output (if not mentioned, then there will be no output to file)
Q__ is the stepover along the travel direction (this is a positive quantity)
R__ (optional) is the retract/pullback amount upon interruption or completion.
P__ (optional) is the interruption PLC bit state which causes a graceful end to the cycle. (If not mentioned, then no PLC bit will checked for graceful interrruption.)
L__ (optional) is the output variable to which to store the interrupt status
(0=no interruption, 1=interrupted by PLC bit P__, 2=surface not found error)

M300 – Fast Synchronous I/O update

There are 32 user definable fast system integer variables that can be used to communicate with the PLC (similar to M94 and M95), but without causing motion to decelerate to a stop* (unlike M94 and M95). The syntax is:

M300 /*nn* /*vvv*

where *nn* is 1-32 and *vvv* is a 32-bit signed integer value. The parameter *nn* (1-32) maps to system variables SV_FSIO1 - SV_FSIO32. These commands work in conjunction with a PLC program that can read the SV_FSIOx and act upon them.

Example:
M300 /21 /-1234 ; set SV_FSIO21 to integer value -1234

* NOTE: Motion will be decelerated to a stop if Smoothing is turned on (P220 = 1).

M333 – Axis Role Re-assignment

This is an experimental M-function that re-assigns X,Y,Z axis behaviors to other axes. This command is not recommended for normal use.

M1000-M1015 – Graphing Color for Feedrate movement

When a CNC program is graphed (F8 from the Main Screen), feedrate movements are normally plotted using the color yellow. This color setting can be changed to another color as stated in the chart below.

M Code	Feedrate Graphing Color
M1000	black
M1001	Navy blue
M1002	green
M1003	teal
M1004	orange
M1005	blue
M1006	lime
M1007	aqua

M Code	Feedrate Graphing Color
M1008	maroon
M1009	purple
M1010	olive
M1011	gray
M1012	red
M1013	fuschia
M1014	yellow
M1015	white

Changing this feedrate graphing color can be used as a method highlighting or hiding parts of a graphed CNC program, but will not affect the normal run of the program (when the CYCLE START button is pressed on the Main Screen). The limitations to using these M codes are as follows: These M codes cannot be placed on the same line as another M code, and also the rapid (G0) movement color cannot be changed.